Automated system for fingerprint authentication using pores and ridge structure

Jonathan D. Stosz
Lisa A. Alyea

Department of Defense
9800 Savage Road
Ft. Meade, MD 20755-6000
301 688-0276

ABSTRACT

A novel technique for automated fingerprint authentication is presented which utilizes pore information extracted from live scanned images. The position of the pores on the fingerprint ridges is known to provide information that is unique to an individual and is sufficient for use in identification. By combining the use of ridge and pore features, we have developed a unique multilevel verification/identification technique that possesses advantages over systems employing ridge information only. An optical/electronic sensor capable of providing a high resolution fingerprint image is required for extraction of pertinent pore information, which makes it unlikely that electronically scanned inked fingerprints would contain adequate pore data that is sufficient, or consistent enough, for use in authentication. The feasibility of this technique has been demonstrated by a working system that was designed to provide secure access to a computer. Low false reject and zero false accept error rates have been observed based on initial testing of the prototype verification system.

Keywords: fingerprint, pore, identification, verification, authentication, recognition, biometric, skeletonization.

1.0 INTRODUCTION

The fingerprint system described in this paper is a member of the family of biometric authentication methods which rely on voice, face, iris, retina, hand geometry, etc. patterns. All biometrics provide an advantage over the use of passwords or PIN numbers alone, in that the user himself must be present. The mode of operation of biometric devices is either verification (live scan to stored template match) or identification (live scan to database of stored templates matching). The template file contains pertinent features, dependent on the biometric technique, and is the result of a one-time, controlled enrollment procedure.

Recently, much effort has been directed towards developing automated fingerprint processing techniques, especially for law enforcement applications. Two primary functions required in fingerprint processing are classification and matching. Classification normally relies on low resolution, or global, characteristics of the ridges. Usually, prints are grouped according to their shape (loop, arch, whorl, or accidental) using a derivation of the nearly 100 year old system developed by Sir E.R. Henry. In contrast, automated matching techniques generally utilize local ridge patterns, or topography, which include ridge end points and bifurcations. The minutiae are defined as medium resolution features within this paper.

In their efforts to modernize fingerprint processing, law enforcement agencies are faced with formidable problems. Generally, classification or matching involves the analysis of large numbers of fingerprints acquired by various, not necessarily compatible, means such as “lifted” latent prints, scanned ink prints from ten print cards, or optoelectronically captured prints from “live scan” devices. Differing fingerprint acquisition methods lead to images with varying characteristics and quality, which compounds the classification or matching problem.

In law enforcement, the push for automation at minimal storage and processing cost has resulted in the emergence of relatively low resolution “live scan” sensors which do not provide reliable or sufficient pore data. Algorithms relying on these devices tend to focus exclusively on ridge structure analysis and consider any pore related features to be noise that should be removed.[1,2,3,4,5] In contrast, the foundation of the work in this paper rests on the belief that pores provide indispensable information, which in combination with ridge structure, creates a more robust overall set of features on which to base an authentication system.

The system described in this paper was designed for the explicit purpose of providing additional computer access security within a controlled environment. Therefore, some of the problems related to law enforcement fingerprint processing do not apply. For

example, a computer security application may involve a relatively small number of valid users and verification of identity is a matching problem, therefore classification of large sets of prints is not needed. Furthermore, the fingerprint image source is a live scan device which provides consistent images that are well adapted to specific processing routines. The optoelectronic, live scan sensor provides a high quality image with resolution sufficient to clearly discern pores, considered high resolution fingerprint features, which are located on ridges.

Interestingly, from a historical perspective, the basis that pores could be used for identification was established as early as 1912 by Locard.[6] Therefore, it is not surprising that in the forensic arena, fingerprint examiners have relied on the analysis of the shape and location of pores to assist in manual fingerprint matching. Based on these facts, it is surprising that automated techniques utilizing pores have not been pursued to a higher degree.

2.0 SENSOR

Analyzing fingerprint ridge patterns and pores requires a system that can produce a high resolution and good quality image of the surface of the finger. The hardware used to acquire this image is collectively referred to as the sensor. The sensor uses a common prism based configuration, [4,7,8,9,10,11] which relies on the principle of total internal reflection.

The sensor incorporates a high resolution CCD video camera and a macro lens which provides the magnification needed to resolve fingerprints and pores. The resulting images are of size 640x480 pixels with 8 bits/pixel. A spatial resolution (sampling rate) of approximately 95 ppmm (2400 ppi) in the horizontal direction and 50 ppmm (1270ppi) in the vertical direction means that a rectangular segment of the finger 6.7x9.6 mm (0.26x0.38 inches) is imaged. The difference between vertical and horizontal resolution results from foreshortening, or the camera's perspective with respect to the print surface, introduced by the sensor.

The sensor provides approximately 2.5 times greater resolution than those conforming to the FBI's current Automated Fingerprint Identification System (AFIS) nominal specifications of 19.7 ppmm (500 ppi) images.[12] A high resolution sensor provides reliable pore information which is not available when using significantly lower resolution devices.

2.1 Print feature size, density, and resolution

Based on measurements made from images acquired using our high resolution sensor, pores may range in size from 60-250 μm in one dimension. Ridges vary in width from 100 μm for very thin ridges to 300 μm for thick ridges. Generally, the period of a ridge/valley cycle is about 500 μm which means there are approximately 14 ridges across an image which usually results in more than 10 minutia points per image. The total number of pores detected within an image is usually at least several hundred giving a pore density of approximately 6 pores/mm^2 (for 385 detected pores/image). The smallest detectable pores, of size 60 μm, determine the minimum resolution requirement for the sensor. Assuming a sampling period half the size of the smallest pore, the corresponding minimum required resolution is approximately 31.5 ppmm (800 ppi).

3.0 IMAGE PROCESSING

The image processing steps, applied prior to feature extraction, are of critical importance in producing a reliable authentication system. Processing reduces degradations in the image that originate from multiple sources and, at the same time, transforms the fingerprint into a form which facilitates the feature extraction process. First, gray scale images from the sensor are converted to binary format. The binary image is stored for later use and then processed further, resulting in a skeleton image. Finally, the skeleton image is processed to improve its functionality from a minutia or ridge analysis viewpoint. During the skeleton processing stage, its quality is improved by eliminating "ridge noise" components produced by pores and also by syntactic processing or "healing." These processing steps are outlined in figure 1.

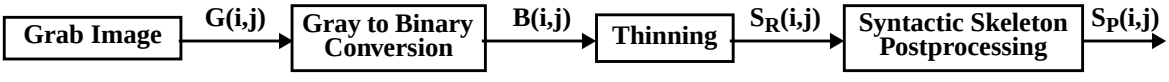

Figure 1. Image processing steps performed prior to feature extraction. The raw image, G, is processed to form the binary image, B, raw skeleton, S_R, and processed skeleton, Sp, which are used for extraction of fingerprint features including pores and minutiae.

3.1 Image degradation

Wrinkles, scars, and excessively worn prints result in images which contain many "false minutia" structures. Also, non-ideal characteristics of an individual's finger, such as relatively oily or dry skin, interfere with the prism based image acquisition and produce false minutia or significant "cloud like" noise (see figure 2). Finally, non-uniform image illumination, dirt, or latent prints on the prism produce unwanted effects and noise which must be removed in order to perform consistent feature extraction.

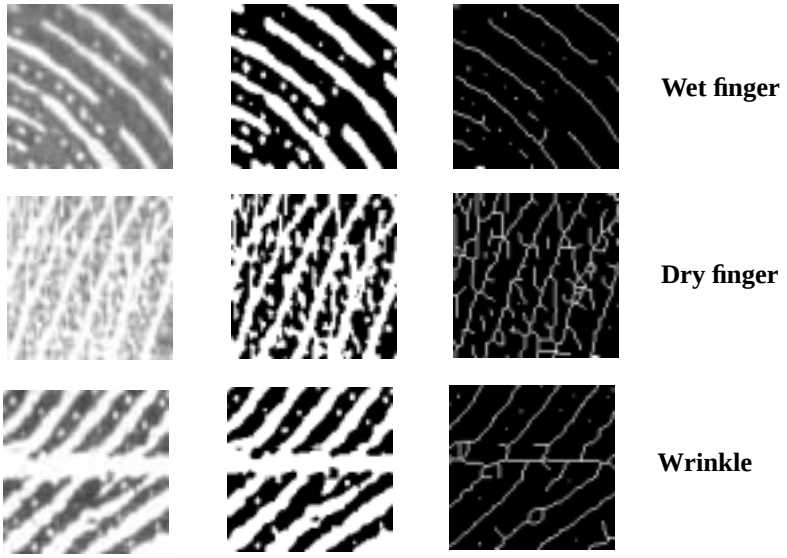

Figure 2. Effects of various sources of degradation on the (a) gray scale, (b) binary, and (c) raw skeleton images. Wet and dry fingers suffer from randomly positioned "false minutia" skeleton objects. Wrinkles and scars create more structured patterns of false minutiae.

3.2 Gray scale to binary conversion

A series of processing routines is used to overcome the undesired effects described above. An initial processing stage uses multiple steps to convert the gray scale image, G, to binary. Figure 3 is a block diagram representation of the gray to binary conversion process.

Figure 3. Gray scale to binary format conversion process.

First, G is low pass filtered by convolution with a 3x3 Gaussian shaped kernel, w.[13]

$$G_{lpf}(i, j) = G(i, j) \otimes w(i, j) \tag{1}$$

The resulting image, G_{lpf}, contains reduced noise but also suffers from smoothing of the high frequency regions of the print such as ridge/valley and ridge/pore transitions. Next, the image is subsampled in the x-direction, by the factor α, to compensate for the difference in vertical and horizontal resolution introduced by the sensor.

$$G_s(i, j) = G_{lpf}(\alpha i, j) \quad \begin{matrix} \text{, for } i = 0, 1, ..., X/\alpha \\ \text{, for } j = 0, 1, ..., Y \end{matrix} \tag{2}$$

Where X and Y are the dimensions of the image. Finally, a dynamic thresholding routine, which compensates for local lighting or finger pressure non-uniformity, converts the image to binary format, B (see figure 3).

$$\mu_{mn} = \left(\sum_{j=nB}^{(n+1)B-1} \sum_{i=mA}^{(m+1)A-1} G_S(i,j) \right) / (AB) \quad \begin{matrix} \text{, for } m = 0, 1, ..., X/A \\ \text{, for } n = 0, 1, ..., Y/B \end{matrix} \qquad (3)$$

Adaptive thresholding is performed by segmenting G_S into X/A by Y/B subregions and then thresholding the subregions separately. X and Y are the dimensions of G_S while AxB defines the size of the local analysis region. The value 19 was selected for both A and B based on heuristic observations of the resulting binary images. Thresholding is done by finding the mean pixel value, μ, given by (3), within a subregion and then converting the pixels with gray level above μ, inside the subregion, to value one and all other pixels to value zero. Pixels with value one and zero are called white and black respectively. The resulting binary image represents the original very well from a morphological perspective with the benefit of being much easier to analyze. Figure 4 shows examples of G_S and B.

(a) G_S (b) B (c) S_R

Figure 4. Preprocessing results. a) G_S is a smoothed, subsampled version of the original gray scale image, G, b) B is the result of dynamic thresholding G_S, c) the result of thinning B to produce the raw skeleton image, S_R.

3.3 Binary to skeleton processing

A skeleton image is produced by eroding the objects within a binary image until they are one pixel wide. It is important to maintain the object's connectivity and ensure that the topography of the original object is well represented by the skeleton. The advantage derived from using a skeleton image is that extraction of ridge features becomes a relatively straight forward procedure based on tracing line segments.

In the binary image, white areas represent both fingerprint valleys and pores while dark areas represent ridges. Our specific interest in the pores leads to the analysis, or thinning, of the white objects on the black background. An iterative, parallel thinning algorithm[14] is used. The application of thinning rules to each pixel over a predetermined number of iterations results in the raw skeleton image, S_R, shown in figure 4.c. The neighbors of a central pixel, p_0, are defined in figure 5, where $\Sigma(p_0)$ is the sum of nonzero neighbors of p_0 and $\tau(p_0)$ is the transition sum of the neighbors of p_0, which equals the number of $0 \Leftrightarrow 1$ transitions in the ordered set $p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_1$.

p_8	p_7	p_6
p_1	p_0	p_5
p_2	p_3	p_4

Figure 5. The central pixel, p_0, and its neighbors.

The thinning rules are:

1) $1 < \Sigma(p_0) < 7$
2) $\tau(p_0) = 2$
3) $p_1 p_5 p_7 = 0$ or $\tau(p_7) \neq 2$
4) $p_3 p_5 p_7 = 0$ or $\tau(p_5) \neq 2$.

Where the sum of the neighbors of element $S(p_0)$ is defined as:

$$\Sigma(p_0) = \sum_{n=1}^{8} S(p_n) \tag{4}$$

and the transition sum of element $S(p_0)$ is:

$$\tau(p_0) = \sum_{n=1}^{9} S(p_n) - S(p_{(n+1)}) \tag{5}$$

S is initially set equal to B, and after a sufficient number of thinning iterations, becomes S_R, the medial axis, raw skeleton representation of B. An ideal skeleton is exactly one pixel wide and centered within the valley or pore. From the skeleton, ends (ridge branch points) and branches (ridge end points) can easily be found by analysis of the neighboring pixels of a skeleton component. End points have only one neighbor and branch points have three neighbors assuming the skeleton is 8-connected.

3.4 Skeleton postprocessing

The raw skeleton image, S_R, may be severely degraded by artifacts resulting from pores, in addition to the processes discussed above. These degradations are substantially reduced by the two skeleton postprocessing steps shown in figure 6. First, the quality of the skeleton is improved by analyzing and then extracting segments representing pores. This process is referred to as "cleaning". In addition, the cleaning routine includes the ability to detect and classify pores. Next, syntactic processing is used to "heal" undesirable artifacts arising from scars, wrinkles, oily or dry skin, etc. and transform the skeleton into a state from which valid minutia information can be extracted. Pore processing and cleaning is done independently of the healing routine. Since the emphasis of this paper is on pores, further discussion will focus on pore processing and cleaning.

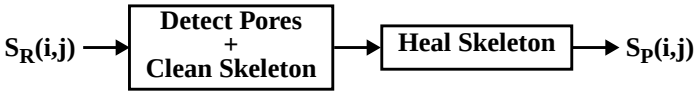

Figure 6. Syntactic skeleton processing methods are used to both clean and heal the skeleton image as well as classify pores and minutia points. The resulting image is the processed skeleton, S_P.

Several commonly encountered skeleton components, or structures, are shown in figure 7. Included in this figure are structures related to pores, minutia points, and undesirable conditions which can be corrected by skeleton postprocessing routines.

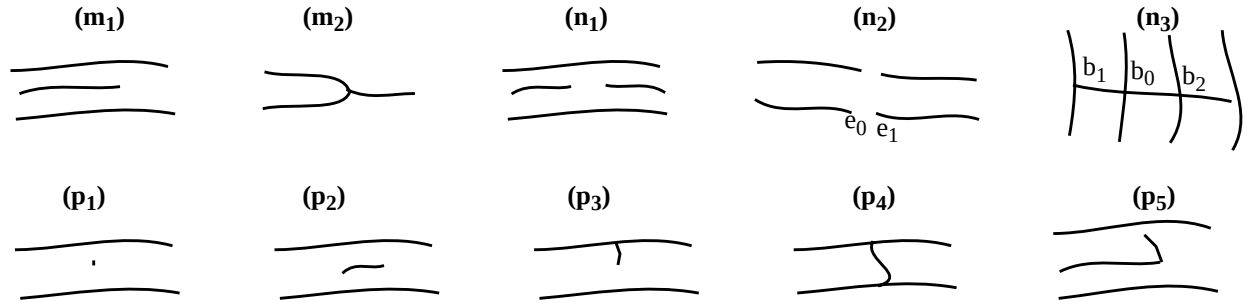

Figure 7. Basic Fingerprint Skeleton Structures - Note: skeleton line segments represent valleys and pores not ridges. m_1 and m_2 are a valid ridge branch point and end point respectively. n_1-n_3 are 'problem' structures which look like valid minutiae. n_1 is a break in a valley which could be caused by dirt, noise, non-ideal thresholding, etc. n_2 represents an area of the print in which valley information has been literally washed away - a typical effect caused by an excessively oily finger for which valleys are filled with moisture. n_3 represents a series of ridges with a wrinkle crossing through them. p_1-p_5 are typical artifacts due to the presence of pores. p_1 is a single pixel, p_2 is a short isolated line segment, p_3 is a short line segment connected to a valley, p_4 is a short line segment connecting two valleys, and p_5 is a short line segment appended to a valley end point.

3.4.1 Cleaning - pore detection

The cleaning routine determines which components of the skeleton represent pores. Once this information is known, the position of each pore is stored, and then all pixels related to the individual pores are pruned from the skeleton. The result is a list of pore locations and an improved skeleton image, S_C. Figure 8.a is an overview of the main steps involved in the cleaning process.

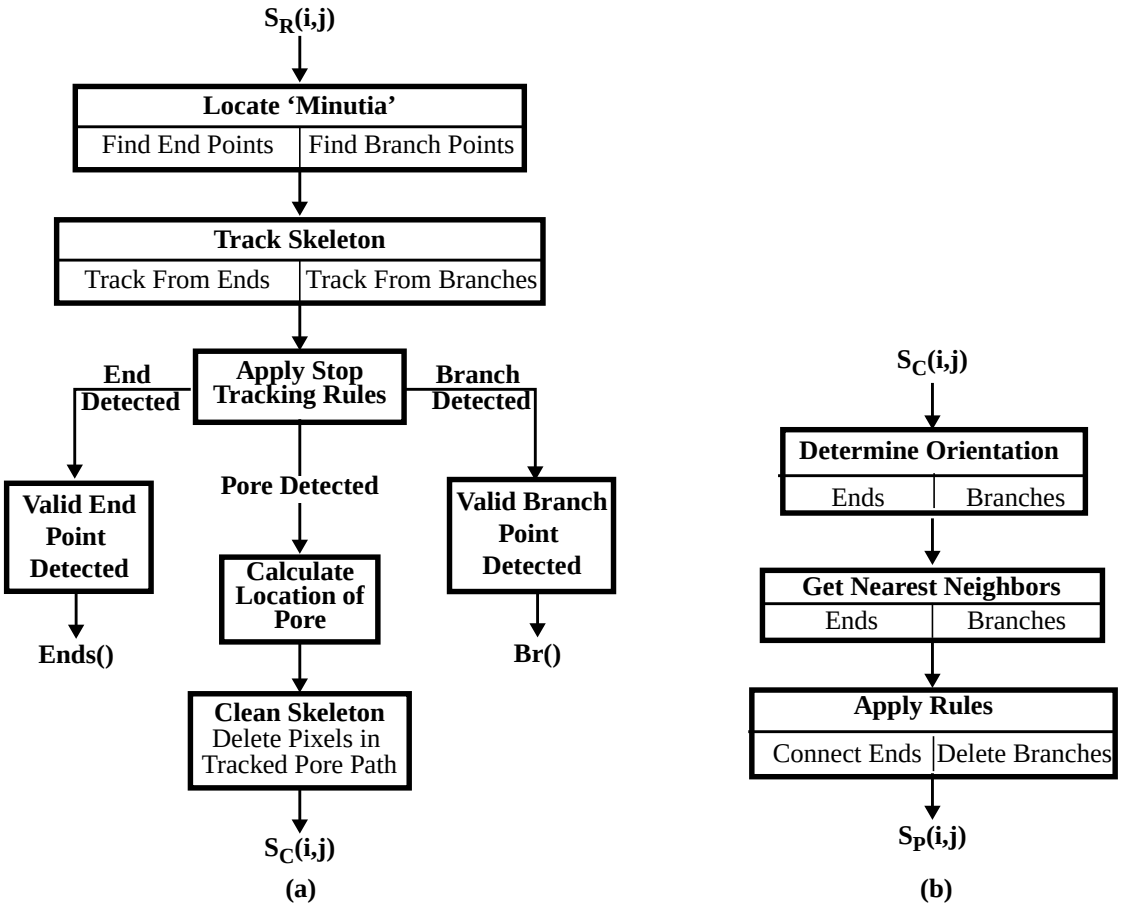

Figure 8. (a) **Detect pores and clean skeleton** - Steps involved in detecting skeleton line segments relating to pores, classifying the pores, and removing pores which cleans the skeleton. (b) **Heal skeleton** - Healing involves detection and correction of skeleton artifacts resulting from scars, wrinkles, or very moist fingers.

3.4.2 Determining end and branch point locations

The first step used to find pores is determining and storing the location of all skeleton end and branch points. It should be noted that skeleton end and branch points are arbitrary terms that do not necessarily correspond to actual print minutia points due to degradations and noise. An end point is defined as any white pixel with either one or no neighbors and a branch point has exactly three neighbors. All other skeleton components have exactly two neighbors and are referred to as connecting points.

There are N_E detected end points with locations denoted by $(x_{E,i},y_{E,i})$ for i = 0,1, ... N_E-1, and N_B branch points with locations $(x_{B,i},y_{B,i})$ for i = 0,1, ... N_B-1. The detected end and branch point locations are used to develop a minutia map which stores pertinent minutia information for later use.

3.4.3 Skeleton tracking and pore detection

Each end point is used as a starting location from which to track the skeleton. Tracking involves analyzing the current element, storing its position as (x_i,y_i) in the path variable P, and determining the location of the next element in the path. Tracking advances one element at a time until a stop condition occurs. The stop conditions for end point tracking are:

1) another end point is detected, or
2) a branch point is detected, or
3) the path length, L_{Path}, exceeds a maximum allowed value.

Conditions 1) and 2) imply that the tracked segment is a pore. Under condition 1), the pore's position, (x_p,y_p), is defined by (6) as the mean value of the coordinates of the elements forming the path of the tracked line segment.

$$x_p = \left(\sum_{i=0}^{L_{Path}-1} P(x_i) \right) / L_{Path} \qquad y_p = \left(\sum_{i=0}^{L_{Path}-1} P(y_i) \right) / L_{Path} \tag{6}$$

Under condition 2), the pore's position, (x,y), is defined by (7) as the first element (starting end point) of the path.

$$x = P(x_0) \qquad y = P(y_0) \tag{7}$$

Condition 3) implies that the tracked segment represents a valid skeleton end point. In this case, the end point location is already known to be $P(x_0,y_0)$ and the orientation is defined to be:

$$\theta = \operatorname{atan}\left(\frac{y_p - P(y_0)}{x_p - P(x_0)} \right) \tag{8}$$

Where x_p and y_p are the mean coordinate values of the elements in the tracked path as defined by equation (6).

The tracking and cleaning procedure continues until all end points have been classified. Then, an analogous procedure of tracking starting at each branch point is done. The stop tracking conditions for branch point tracking are:

4) another branch point is detected, or
5) an end point is detected, or
6) the path length (of all tracked branches) is greater than the maximum allowed path length.

If condition 4) is satisfied, a pore is detected and its position is again the center of mass of the line segment connecting the two branch points. Condition 5) should have already been taken care of by end point tracking. Condition 6) implies the presence of a valid skeleton branch point. The location is already known and the orientation is defined as the angle of the branch which is most separated in angle from the other two branches using equation (8). Tracking, classification, and storage of end points, branch points, and pores produces both a minutia and a pore map which hold the information needed for feature extraction.

3.4.4 Pore pruning and minutia classification

Once the location of a pore has been established, the elements contained in its path are pruned from the skeleton by converting them to black pixels. At the same time, any branch or end points associated with the pore are removed. Therefore, after all the cleaning has been done, the remaining branch and end points correspond to valid print features. Their location, type, and orientation represent the minutia map of the print. Similarly, the location of detected pores defines the pore map of the print. The reliability of the pore detection process is demonstrated in figure 9 which also shows an improved, "cleaned" skeleton. End point tracking enables the pruning of pores and classification of the structures m_1, p_1, p_2, p_3, and p_5 from figure 7 corresponding to a valid end point and pores respectively. Branch point tracking enables the pruning of pores and the classification of structures m_2 and p_4, from figure 7, which are a valid branch point and pore respectively.

3.4.5 Healing the skeleton

Cleaning is based on the analysis of a single connected skeleton object. If an object conforms to a set of constraints, it is defined as a pore and then classified and removed. In contrast, healing requires a higher level of analysis, since generally, multiple neighboring skeleton segments are disrupted by a degradation process. Figure 2 provides actual examples while these effects are illustrated by structures n_2 and n_3 in figure 7. For example, in order to determine that structure n_3 is a result of a wrinkle, and that each component is not by itself a skeleton branch point or pore, the analysis of the whole disrupted local region is required.

Similarly, healing of structure n_2, requires analysis of multiple skeleton segments in order to determine that the segments should actually be joined together.

Healing of the structures n_1, n_2, or n_3 is accomplished by finding the position and orientation of all branch and end points of the cleaned skeleton, see figure 8.b. In addition, for each skeleton minutia point, the position of the nearest several neighbors is determined. By analyzing this combination of information, a certain degree of healing of the skeleton can be done.

(a) (b) (c)

Figure 9. Pore Detection - The ability to reliably detect the position of pores is demonstrated in (a) where detected pore locations are denoted by a square box. The effect of "cleaning" the skeleton is seen in (c) as compared to (b) which is the raw skeleton image. Note: Pore detection and cleaning of the above images was based on end point tracking only - no branch point tracking was done.

3.4.6 Healing broken skeleton segments

Two disconnected skeleton line segments can be connected, or healed, if they satisfy a set of constraints.

1) the distance between end points e_0 and e_1 (see n_2 in figure 7), must be less than a maximum distance threshold value.
2) the orientation of end points e_0 and e_1 is such that the line segments, used to determine their orientation, must be pointing at each other within a maximum difference angle threshold value.

If these two conditions are satisfied, the line segments are connected together by adding a straight line of pixels between points e_0 and e_1 in the skeleton.

3.4.7 Healing wrinkles

Differentiating between the valid pore structure, p_4, of figure 7 and the structure n_3, which represents a wrinkle, can be done by analyzing information on neighboring branch points. The horizontal line segments can be removed from structure n_3 if the following conditions exist:

1) a branch point, b_0, has at least two neighboring branch points, b_1 and b_2, which are each no further away than a maximum distance threshold value.
2) the orientation of one branch of b_1 must be 180 degrees, plus or minus a threshold maximum angle difference, different from the orientation of one branch of both b_0 and b_2.
3) these branches must also point at each other or be closely connected on common line segments.

4.0 ENROLLMENT

Figure 10 shows a block diagram of the enrollment procedure, which consists of fingerprint image acquisition, preprocessing, feature extraction, and creation of a user specific template file which contains pertinent authentication information. Image acquisition and preprocessing result in binary and skeleton representations of the print which are both used for feature selection and extraction.

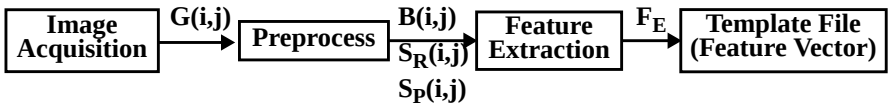

Figure 10. Enrollment - main steps.

By examining the binary image, several small regions of the print are selected, extracted, and saved in the template. Each region is selected manually and includes interesting fingerprint features. The segment selected first is referred to as the origin locator segment (OLS). The location of each other selected data segment is stored in the template as an offset x and y value, (Δx,Δy), from the OLS. Generally, the OLS includes the core of the fingerprint pattern while other segments include minutia points. Depending on the print quality and feature (minutia and pore) density, four (poor quality print) to ten (good quality print) segments are selected from the print and stored as packed, binary data in the template file. The size of the OLS is fixed as 64x64 pixels, other segments may vary in size ranging from 24x24 to 64x64 pixels according to how many segments are used.

Only minutia and pore information contained within the selected local regions of interest is used for authentication. As a result of skeleton processing, all detected pore locations and minutia point characteristics (type, location, and orientation) are predetermined and temporarily saved. When the position of a local region of interest has been selected, the minutia data and pore locations associated with the segment are recalled and stored in the template file. Figure 11 gives a graphical description of the feature selection, extraction, and storage components of the enrollment procedure.

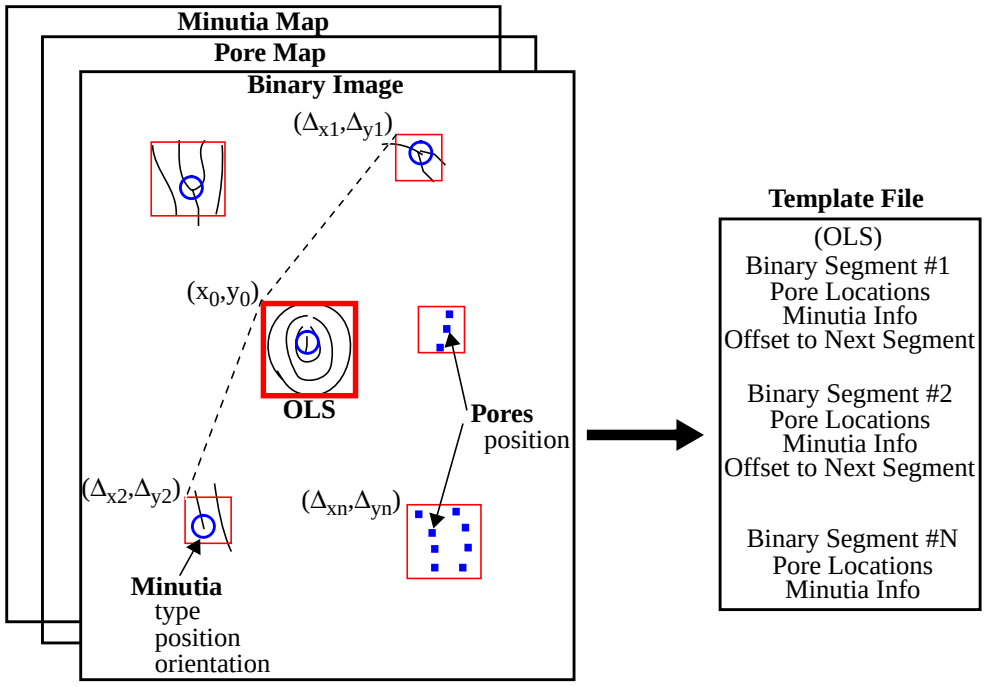

Figure 11. Enrollment - selection and extraction of features used for authentication. The template consists of three primary types of data: binary image segments, detected pore locations, and minutia information. As a result of preprocessing, the pore map and minutia map contain all of the pertinent information about pores and minutia points which is readily saved to the template file for a small region of interest once it is chosen.

5.0 VERIFICATION

Verification is a form of authentication involving a one-to-one match between a live scan and the user's template. The verification steps are shown in figure 12. A user provides the name of his template, T, to the system, the information is retrieved, and then

matched against a live scan of the user's print. Verification involves a comparison of the template's binary data segments, minutia

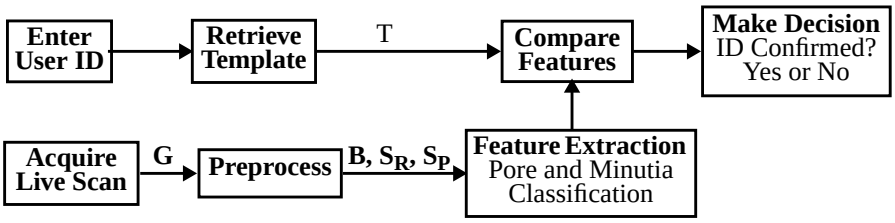

Figure 12. Main verification steps.

characteristics, and detected pore locations to the processed live scan image (see figure 13.a for details). Similar to enrollment, verification requires gray scale to binary and binary to skeleton preprocessing. Matching is initiated by determining the origin of the live scan print which is defined as the position of maximum correlation between the OLS and the processed live scan binary print. Next, the remaining binary image segments from the template are compared to the live scan. Their stored offset locations from the OLS are used to estimate the best matching position, which is determined more precisely by searching a small region around the estimated location. A segment matching score, $S_{S,i}$, results for each image segment, i, used in the template. An overall score, which combines the individual segment matching scores, is defined by (9) and used in the verification decision.

$$S_S = \left(\sum_{i=0}^{N_S - 1} S_{S,i} \right) / N_S \tag{9}$$

Where N_S is the number of image segments used. S_S is normalized in the range of [0,1] with a score of 1.0 corresponding to a perfect pattern match. S_S represents an average measure of how well the template segments match the live scan print and is independent of minutiae and only slightly dependent on the presence of pores within the segment.

5.1 Scale, translation, rotation, and plasticity

For fingerprint authentication, a fixed sensor device eliminates image scale variation, while correlation matching inherently adjusts for changes in alignment or translation of the finger. There are still two types of image variation which cause problems that must be addressed.

First, plasticity of the finger is a trait which may cause substantial variation of the relative position and orientation of fingerprint features on a global scale. The degree of variation is relatively small although, when observing local regions of the print, which is the reason for using small image segments. Also, by allowing for variation in the size of the search area for matching the segments, small changes in the relative position of the print features are overcome.

Second, the problem of image rotation must also be addressed. Overall image rotation can result from placement of the finger at an angle which differs from the enrolled print, or local regions can be rotated relative to one another due to the plasticity of the finger. Rotation causes a variation in the relative position and orientation of the print features. This problem is compensated for by rotating the template's binary segments over the range of ±10°, in 2° increments, and then comparing each of the rotated states to the corresponding local area of the live scan image. The state which matches best provides an estimated angle of rotation of the live scan segment relative to the template segment. The angle of rotation is then used to modify the stored offset value which estimates the position of the next image segment. These modifications are intended to correct for uniform global print rotation. Once the relative angle of rotation between the stored template segments and the corresponding live scan segments is known, the expected location and orientation of the minutia points and the location of the pores can also be modified.

5.2 Multilevel verification

The binary image segments from the template are used to locate desired regions of analysis of the live scan print in addition they provide an image segment matching score. Our system is termed multi-level because the medium and high resolution features, including minutia data and pore locations, are only evaluated if the segment matching score for a particular segment exceeds a defined threshold value, T_S. If a given segment matches the live scan well enough, then the template minutia and pore information corresponding to that live scan segment are compared, resulting in a minutia match score, S_M, and a pore match

score, S_P. Figure 13 provides a description of the matching procedure including the multi-level concept. A user's identity is confirmed only if:

$S_S \geq T_S$, and
$S_P \geq T_P$, and
$S_M \geq T_M$.

Where T_S, T_P, and T_M are the predefined threshold values for the segment, pore, and minutia matching scores respectively.

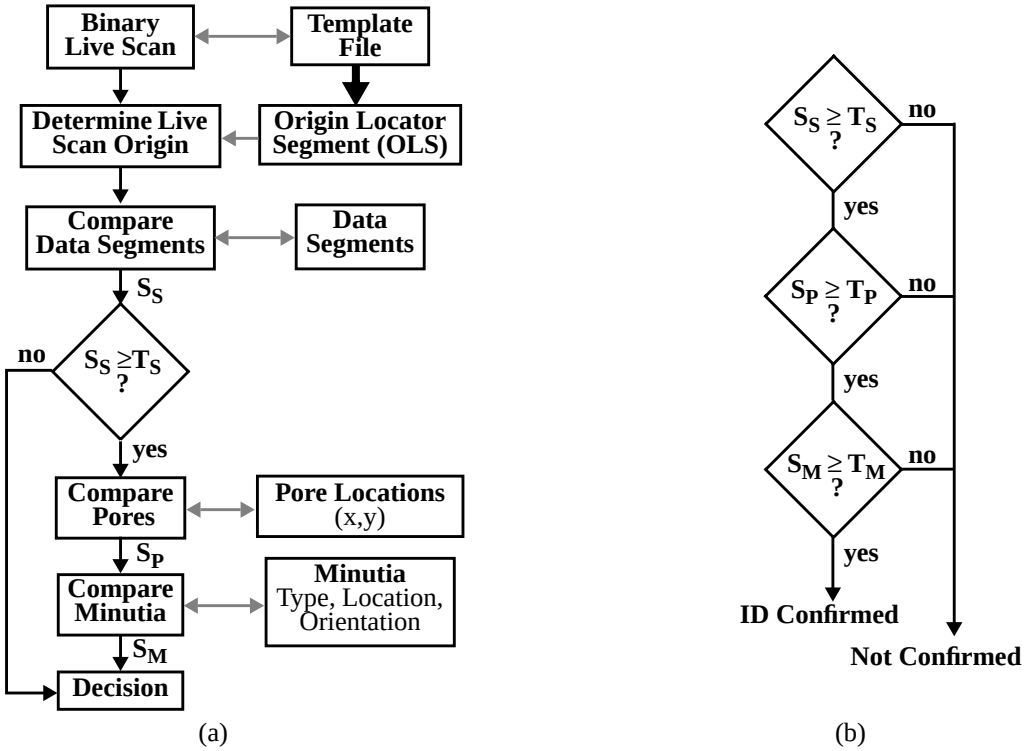

Figure 13. Multilevel verification - a) main feature matching steps where live scan parameters and verification procedures are on the left side of the block diagram and stored template information needed for verification is located on the right side and b) decision process involving data segment, pore, and minutia matching results.

5.3 Pore and minutia matching

The type, position, and orientation of minutia points and the location of pores, found within a template binary segment, are stored in the template file. Preprocessing of the live scan image results in the detection and classification of its minutia points and pores. Therefore, after determining the origin of the live scan image, so that it corresponds to the origin of the template image, the location of features from the same finger will be in alignment except for variations in position due to rotation and finger plasticity. The features can then be compared to check for a match.

The template features related to pores arise only from pores located within the image segments extracted from the binary image during enrollment. The first step in verification is alignment and scoring of these segments with the live scan image. Once alignment has been done, pore matching consists of determining if live scan pores exist at specific positions stored in the template file. The pore matching score, S_P, is the ratio of pores registered in the enrollment session to the number of pores confirmed during verification.

$$S_P = \left(\sum_{i=0}^{N_S - 1} N_{MP,\,i} \right) \Big/ \left(\sum_{i=0}^{N_S - 1} N_{P,\,i} \right), \tag{10}$$

where $N_{P,i}$ is the number of pores detected in template segment, i, and $N_{MP,i}$ is the number of matching pores in segment i.
The type, location, and orientation of the minutia points stored in the template are then compared to minutia information from

the live scan. A minutia point match is considered to be made if the type and orientation of the template minutia matches a live scan minutia in the correct location. Type matching is a binary decision (end point or bifurcation), whereas location and orientation allow for some variation. The constraint on location is simply that a minutia point from the live scan must be located within the image area corresponding to the template binary segment. The orientation constraint is that the detected minutia angle of rotation must be within 45° of the corresponding template minutia orientation value. The minutia matching score, S_M, is the ratio of correctly matched minutia points to the total number stored in the template file.

$$S_M = \left(\sum_{i=0}^{N_S - 1} N_{MM,\,i} \right) \Big/ \left(\sum_{i=0}^{N_S - 1} N_{M,\,i} \right), \tag{11}$$

$N_{M,i}$ is the number of minutia classified in template segment, i, and $N_{MM,i}$ is the number of matching minutia in the corresponding live scan region.

6.0 TESTING RESULTS

Tests were conducted on two databases of fingerprints to establish false accept (FAR) and false reject (FRR) error rates. The first database contained 258 different fingerprints acquired from 137 individuals. The subjects ranged in age from 8 to 60 years and consisted of 60% males. The database distribution of 63.5% loops, 4% arches, and 32.5% whorls or accidentals corresponds statistically well to that of much larger reported databases. The second database contains multiple images of one finger from each of 10 individuals (7 males). Three images per session were acquired from each subject over a period of 8 weeks with at most two sessions per day. The first image from each session represented the user's attempt to provide a consistent day to day print. While on the second and third acquisitions, the subject was asked to vary the amount of finger pressure, moisture/oil content on the skin surface, and/or skin temperature in an attempt to vary the print characteristics. In order to maximize the number of usable test images, the subjects were allowed to view their fingerprint on a video monitor in order to achieve consistent alignment.

The quality of a fingerprint image can be defined in terms of both its potential and consistency for use in authentication. Potential is assessed based on the number of minutiae and pores available in the print. For example, a print with plentiful minutia and pore features rates as high potential. The consistency of an image is dependent on noise and degradations (in particular those related to skin moisture content). Each fingerprint in the database was rated on potential (pore and minutia density) and consistency (noise content) by assigning a score ranging from 0 (excellent) to 2 (poor) for each of the three parameters. A cumulative score, defined as the sum of the three individual scores, represents the overall image quality. The range of cumulative scores is from 0 to 6. Good quality prints were defined to be in the 0-1 range and constituted 53.3% of the images. Medium quality prints were defined in the 2-4 range and constituted 41.7% of the images. Finally, bad quality prints (5-6 range) made up 5% of the database. Both the individual and total scores were derived in order to determine the effect of these parameters on verification error rates.

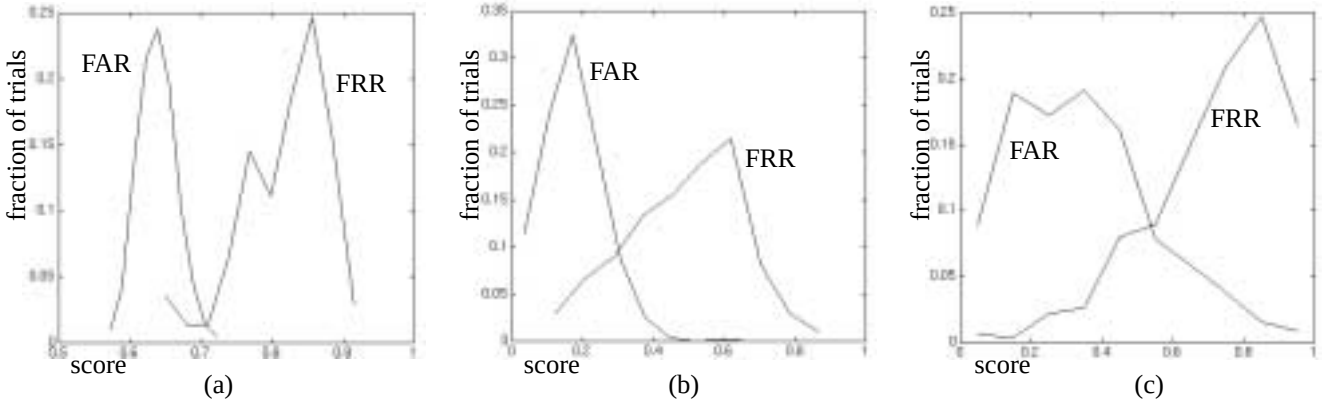

Figure 14. Normalized histograms of scores for independent matching of the three features used for authentication with FAR (enrolled users matched against all other prints in the database) and FRR (templates from given users matched against multiple images of those users). a) data segments (FRR: mean = 0.819, std = 0.061; FAR: mean = 0.638, std = 0.027), b) pores (FRR: mean = 0.487, std = 0.161; FAR: mean = 0.172, std = 0.087), and c) minutiae (FRR: mean = 0.734, std = 0.184; FAR: mean = 0.354, std = 0.209).

In c he

verification algorithm was adjusted, resulting in separate matching scores for all three features. Histograms of the scores were calculated and are shown in figure 14. Figure 15 shows plots of the FAR and FRR when assuming that the system relies on any one of the three features alone. The independent equal error rates (EER) of 0.048, 0.120, and 0.171 for the three authentication features correspond to threshold values of: $T_S = 0.685$ (for data segments), $T_P = 0.275$ (for pores), and $T_M = 0.569$ (for minutiae) respectively. Based on the plots in figure 14, it is evident that the separation between matching scores for FAR and FRR type testing is sufficient for any one of the features to be used as a basis for an authentication system.

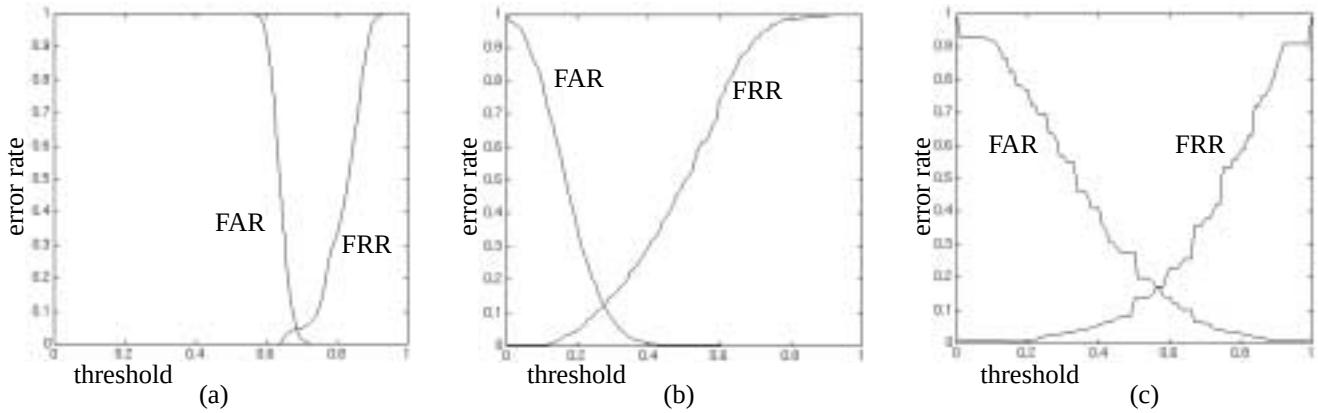

Figure 15. Error rates for the independent matching of three types of features used for authentication a) data segments, b) pores, and c) minutiae. FRR (false reject rate) and FAR (false accept rate) curves are given verses threshold value.

The performance of a system relying on all three fingerprint features depends on the feature matching scores and their respective decision threshold values. In addition to the three matching scores, S_S, S_P, and S_M, and their associated threshold values, T_S, T_P, and T_M, another important parameter is defined as the number of raw data segments from the template that are located within the bounds of the live scan. This parameter is assigned a score, S_{in} (fraction of segments in bounds) and a threshold decision value, T_{in}. The precedence for a correct template to live scan match is: first, $S_{in} \geq T_{in}$, then $S_S \geq T_S$ and $S_P \geq T_P$ and $S_M \geq T_M$. Note that the data in figure 14 and figure 15 reflects matching scores which are based upon the number of segments found in bounds, while segments that are out of bounds do not contribute to this score. Generally if T_{in} is implemented and declared to be some realistic value, then the FRR's as represented in figure 15 should decrease, since the origin of an incorrect finger is usually located such that a significant percentage of the data segments are not found within the bounds of the image.

The threshold values and error rates can be written in vector form where, $\mathbf{T} = (T_S, T_P, T_M, T_{in})$ and $\mathbf{E} = (E_{FRR}, E_{FAR})$. The improvement in error rates for a system using multiple features can be demonstrated by adjusting the threshold values for each feature. For example, if $\mathbf{T} = (0.685, 0, 0, 0.5)$ then $\mathbf{E} = (0.0497, 0.0425)$ for a system relying only on data segment matching. If pores and minutia are included with $\mathbf{T} = (0.685, 0.17, 0.35, 0.5)$, then $\mathbf{E} = (0.0716, 0.0105)$. In this case, the FAR has been reduced by a factor of 4.04 at the cost of an increase in the FRR by a factor of only 1.44. As another example, if $\mathbf{T} = (0.715, 0.38, 0, 0.5)$ then $\mathbf{E} = (0.2505, 0)$ and if $\mathbf{T} = (0.715, 0, 0.67, 0.5)$ then $\mathbf{E} = (0.3161, 0)$. This result is important because it implies that the pore matching technique is more effective than minutia matching for reducing the FAR. Finally, if $\mathbf{T} = (0.72, 0.16, 0.05, 0.5)$ then $\mathbf{E} = (0.0696, 0.0004)$ which demonstrates that a negligible FAR and a relatively low FRR can be achieved by combining data segment, pore, and minutia features.

7.0 CONCLUSION

A description of a novel live scan fingerprint authentication system which utilizes pore information has been presented. It has been proposed that current commercial live scan sensors do not provide sufficient resolution to utilize the pore information in a reliable manner. Furthermore, it has been proposed that an added measure of security may result from the inclusion of pores in a verification system. Test results conclusively establish the viability of using the pores in combination with minutiae and ridge structure for authentication. Based on the results of the work presented in this paper, further research on the use of pores in authentication systems is warranted.

8.0 REFERENCES

1. T.Ch.Malleswara Rao, “Feature extraction for fingerprint classification,” Pattern Recognition, Vol 8, pp 181-192, 1976.

2. L. Coetzee, E.C. Botha, “Preprocessing of two-dimensional fingerprint images for fingerprint recognition,” COMSIG 1991 Proceedings, South African Symposium on Communications and Signal Processing, pp 69-73, IEEE, 1991.

3. K. Fukue, H. Shimoda, T. Sakata, and B. Kim, “Fingerprint verification system - verification algorithm,” Proceedings of TENCON 87: 1987 IEEE Region 10 ‘Computers and Communications Technology Toward 2000’ in Seoul, Korea 25-28 Aug. 1987, Vol 1, Korea Institute of Electronic Engineering, Ministry of Communications, et al. pp 71-75,New York: IEEE.

4. B. Kim, C. Kim, and T. Sakata, “Automatic fingerprint verification system: Digital image processing algorithm,” Proceedings of TENCON 87: 1987 IEEE Region 10 ‘Computers and Communications Technology Toward 2000’ in Seoul, Korea, 25-28 Aug. 1987, Vol 1, Korea Institute of Electronic Engineering, Ministry of Communications, et al. pp 66-70, New York: IEEE.

5. A. Luk, S.H. Leung, C.K. Lee, W.H. Lau, “A Two-Level classifier for Fingerprint Recognition,” 1991 IEEE International Symposium on Circuits and Systems, pp 2625-2628, IEEE, 1991.

6. E. Locard, “Les Pores et L’Identification Des Criminels,” Biologica: Revue Scientifique de Medicine, Vol 2: 357-365, 1912.

7. Z. Chen and C.H. Kuo, “A Topology-Based Matching Algorithm for Fingerprint Authentication,” Proceedings, 25th Annual 1991 IEEE International Carnahan Conference on Security Technology, pp 84-87, IEEE, 1991.

8. K.H. Fielding, J.L. Horner, C.K. Makekau, “Optical Fingerprint Identification by Binary Joint Transform Correlation,” Optical Engineering, Vol 30, No 12, pp 1958-1961, Dec. 1991.

9. M. Kawagoe and A. Tojo, “Fingerprint pattern classification,” Pattern Recognition, Vol 17, No 3, Great Britain: Pergamon Press Ltd., pp 295-303, 1984.

10. K. Morita and K. Asai, “Automatic fingerprint identification terminal for personal verification,” Hybrid Image Processing, SPIE 638, pp 174-181, 1986.

11. M. Takeda, S. Uchida, K. Hiramatsu, and T. Matsunami, “Finger image identification method for personal verification,” Proceedings of the 10th International Conference on Pattern Recognition in Atlantic City, New Jersey 16-21 June 1990, Vol 1, International Association for Pattern Recognition. Los Alamitos, Calif.: IEEE Computer Society Press, pp 761-766.

12. Requirement described in the draft ANSI standard: Data Format for the Interchange of Fingerprint Information, dated. June 17, 1993.

13. P. J. Burt and E.H. Adelson, “The Laplacian pyramid as a compact image code,” IEEE Transactions on Communications, Vol COM-31, No 4, pp 337-345, April 1983.

14. R. Stefanelli and A. Rosenfeld, “Some parallel thinning algorithms for digital pictures,” J.ACM, Vol 18, No 2, pp 255-264, 1971.