Slap Fingerprint Segmentation

Puneet Gupta and Phalguni Gupta
Computer Science and Engineering Department
Indian Institute of Technology Kanpur
Kanpur-208016, India
{puneet,pg}@cse.iitk.ac.in

Abstract

This paper proposes an efficient algorithm to extract all four fingerprints from a slap-image and to identify the indices of these fingers such as index, middle, ring or little finger of left or right hand. Entropy of the image is used to determine different components of fingers, while geometrical and spatial properties of these components are considered to segment fingerprints from the slap image. The algorithm gives accurate results even when there exist some dull prints, large rotational angles or non-elliptical shape of components in the image. It makes use of a new distance measure to index these fingerprint components. It has been tested on a database of 6,096 images of 1,016 subjects, which has accurately segment each slap image into four fingerprints.

1. Introduction

One of the major problems in today's digital world is to design an accurate but automatic personal authentication system in near-real time. Fingerprint is an extensively studied biometric trait because it exhibits characteristics like uniqueness, permanence and difficult to forge. It has been pointed out in [7] that identification accuracy can be increased if one can use multiple fingerprints for authentication, instead of single fingerprint. There exists slap fingerprint device which can be used to acquire the slap image of multiple fingers of a hand. However, multiple fingerprints can also be acquired by capturing single fingerprint for each finger. But one can minimize the problem of forgery if the slap fingerprint device is used to acquire the image. Also it takes less amount of time [6]. It has been observed that such a device can capture multiple single fingerprints instantaneously without much operator intervention.

For fingerprint authentication, the slap fingerprints are required to be segmented to extract features from each fingerprint and perform matching. This process of separating individual fingerprints from a given slap image is known as "slap fingerprint segmentation" . Beside the good feature extraction and matching techniques, the performance of any personal authentication system using slap-fingerprint device is dependent on segmentation technique. It requires to satisfy certain constraints such as:

1. *Accurate*: All fingertips in slap fingerprint are correctly detected and the fingerprint images that are segmented out, do not contain other fingerprint regions.
2. *Classifying hand type*: It should be able to correctly recognize whether the slap fingerprint image belongs to right hand or left hand.
3. *Classifying each finger type*: Each fingerprint image obtained from the slap image must belong to an image of one of the four fingers viz., index finger, middle finger, ring finger or little finger.

This paper proposes an efficient and rotation invariant segmentation algorithm to segment all finger images from the slap fingerprint image. The proposed algorithm satisfies all the above constraints. It works efficiently even for a large angle of rotation, for dull prints in small portion of the image and for non-elliptical shape of fingerprint image. It is organized as follows. Next section discusses various available methods of a four slap fingerprint segmentation. The proposed method has been proposed in Section 3. Experimental results have been given in Section 4. Conclusions are given in the last section.

2. Literature Survey

The method in [6] to segment a slap-image into fingerprints has downsampled slap fingerprint image by averaging 8x8 blocks of pixels and has binarized using the mean pixel value as threshold. Narrow ranges of angles are considered and parallel lines are drawn oriented at these angles at some heuristically chosen points. Count of black and white pixels are separately maintained along each parallel line to form two histograms where each parallel line represents a bin and black pixel count and white pixel count as

978-1-4673-1228-8/12/$31.00 ©2012 IEEE

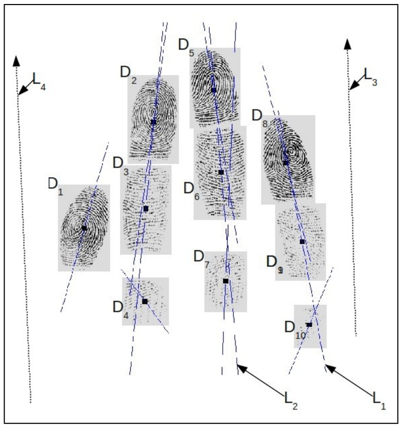


Figure 1. Finger Components with Local and Global Orientations

observations in the histograms. Local maxima for the histogram corresponding to black pixels gives the centers of fingers, while that corresponding to white pixels provides the boundary between two fingers. The best angle is chosen in such a way that different fingers are separated from each other and fingers corresponding to this angle are marked as true fingers. From each actual finger, fingerprint is extracted out using heuristics of a finger. Prior knowledge of capturing device often gives better results because one can predict the desired line spacing and orientation. It uses multiple passes based on angles and number of lines and as a result, it is computationally expensive. Another observation is that all fingers are assumed to be in a common direction which may not be always the case as shown in Figure 1.

In [3], a slap-fingerprint image is binarized and splitted into disjoint components. Edge detection is applied on this preprocessed image to estimate convex hull and orientation for each component. Based on relative orientation and placement of disjoint components, some splitting and joining heuristics are applied to form hand geometry and to avoid merging of different fingers. These components are used to obtain the segmented fingerprints. This method works well for small degree of rotation.

The segmentation technique proposed in [2] has assumed that the fingertips are elliptical in shape. A two-staged mean shift is applied to find locally dense clusters or finger components. It has used ellipse-fitting algorithm twice to prune problematic components by computing mean orientation and size from the observed components. Mean orientation is assumed to be the hand orientation, called global orientation. Slap image is rotated in the direction of global orientation, i.e, slap image is turned upright. All components belonging to a finger except the one containing fingerprint are assumed to be placed below the component containing fingerprint. These fingerprint components are given unique labels based on their corresponding location in the left or right hand. Fingerprints are labeled based on the heuristic that middle finger is the longest one. This heuristics helps to determine whether the slap image represents left or right hand image. Some prior knowledge about image and capturing device is useful for getting better results. Global orientation of a hand is estimated with the help of mean orientation, which may not be a good approximation always. One of the most crucial factors is that this algorithm is computationally expensive due to the use of mean-shift algorithm. But this effect has been minimized by reducing the original image size. Further, mean-shift is applied by using the assumption that fingertips are elliptical in shape which is not always applicable.

In [9], a better technique for noise removal and foreground segmentation have been presented. The image is down-sampled and mean and variance are calculated from this image locally on a block of size 8×8 at each pixel. Based on these means and variances, some pixels are considered as background pixels. Image is adaptively binarized and local ridge frequency is calculated by estimating the maxima in Fourier magnitude spectrum within a block and if this maxima of a block is above certain threshold, then the block is considered as foreground. The local frequency calculation is useful in more finer segmentation, but is computationally expensive. These blocks are merged to form a component when the distance between them is smaller than a threshold. From these components, local orientation is estimated using ellipse fitting proposed in [2] and knuckle lines are extracted for extracting fingerprint from each of these components. Global orientation of hand is calculated using weighted average where weights are assigned based on the orientation and area of the components. This may not be a good approximation always. To illustrate this, let us consider the image shown in Figure 1. Local orientation of each component is shown with dotted lines passing through the components. The methods proposed in [2] and [9] has been used to obtain the direction of global orientation of the image which are shown with the help of lines L_3 and L_4. Center of each component is then projected onto the line perpendicular to the global orientation of the hand and if distance between two component projections is small, then these components are considered to belong to a finger and the top one is chosen as the fingerprint image. That means, components belonging to a finger have centers closed to a line, whose orientation is similar to the orientation of that finger. This finger orientation is approximated by global orientation. But when fingers have different orientations as shown in Figure 1), then this approximation fails due to which results may suffer. Finally, confidence of each component consisting of fingerprint is calculated heuristically with the help of semi-minor axis, semi-major axis and area of the component. Prior knowledge about image is found to be useful for setting these thresholds. Global orientation formed by weighted average of local orientations is not always approximated accurately; as a result, fingerprint extraction from components may be inaccurate. This algorithm gives much better results than [2], as pointed out in [8].

In [8], the technique used in [9] has been modified by estimating principal axis for each block by minimizing the cost function based on rotational inertia with respect to the center of gravity of the slap image and crease is detected based on heuristics. Improvement in the speed and accuracy has been observed.

Each of these well known algorithms has some limitations which are mentioned below:

1. Shape of component containing fingerprint is assumed to be elliptical.

2. Global orientation of hand is approximated using local orientation which is used for segmenting fingers and finding the hand type.

3. Local orientation of a finger is approximated using global orientation for segmenting fingers.

4. These are rotationally invariant only for small angle deviation.

5. Different fine-tuned thresholds are used.

6. Many algorithms do not give accurate results when there exist dull prints in a small region within a fingerprint.

3. Proposed Method

This section presents the proposed method to extract fingerprints from the slap-fingerprint. It consists of two major steps. In the first step, it segments fingerprint components and relationship among these components are used to determine the hand type - right or left. Finally, each finger has been classified into one of index, middle, ring and little finger. Component detection is not based on any constraint on the shape of the fingerprint components. Some geometrical and spatial properties of each of these components are analyzed to merge components for finger detection. It has not considered global orientation of the hand and local orientation of the fingers. The algorithm is found to be rotationally invariant. It is also robust against the presence of dull prints and non-elliptical shape of fingers. However, it assumes the existence of all four fingers in the slap image.

3.1. Component Detection

This subsection proposes the method to detect finger components from a given slap image. It can be noted that the size of slap image is very large and it is difficult to segment this image in near real time. Bicubic interpolation is used to downscale the image into one-sixteenth of its original size. To remove salt and pepper noise, median filter is applied on the image. Some finger components may be dull, as shown in Figure 1. This may be due to uneven pressure, smooth fingers or dry-wet fingers. As a result, some time it is difficult to detect such type of fingerprint components with the help of traditional binarization technique.

In this paper, we have used entropy based technique to segment all fingerprint components. Entropy can be defined as the dispersion of the probability distribution [4] which can be calculated by counting the number of times each intensity i occurred in the image and normalizing this by dividing it with the total size of the image. Let p_i refers to the probability distribution corresponding to the i^{th} intensity bin. Shannon entropy [4] which is given by

$$H = \sum_i p_i log \frac{1}{p_i} = -\sum_i p_i log p_i \tag{1}$$

This Shannon entropy is calculated on each pixel around it's 3×3 neighborhood. Image with a single gray-scale value has a low entropy value because of its less dispersion. Each finger component has higher entropy, while other part do not have much dispersion in intensity level leading to low entropy. A dull fingerprint component contains a significant entropy.

For a slap image I, its corresponding entropy image E consists of several connected components which are labeled based on 8-connected neighbors. Some components are formed due to some noise in the background. These components have relatively smaller area and are eliminated if their respective area is less than certain threshold T. Contextual knowledge of slap fingerprint image has been used to determine such a threshold.

Let $C_1, C_2, ..., C_n$ be the components found from E, after eliminating all components having smaller area. Let (x_{C_i}, y_{C_i}) be the center of the component C_i where

$$x_{C_i} = \frac{\sum_{i=1}^{p} x_i}{p} \qquad y_{C_i} = \frac{\sum_{i=1}^{p} y_i}{p} \tag{2}$$

Principal component analysis (PCA) [5] is used to estimate orientation of each component. Unbiased estimate of variance-covariance matrix, A_{C_i}, is formed using pixels belonging to C_i. Eigenvalues λ_a and λ_b are computed by solving the characteristic polynomial of A_{C_i}. Eigenvector e corresponding to the larger among λ_a and λ_b gives the estimation of orientation θ_i of C_i where θ_i is given by:

$$\theta_i = \tan^{-1}\left(\frac{e(2)}{e(1)}\right) \tag{3}$$

where $e(1)$ and $e(2)$ are the projections of e along x and y axis.

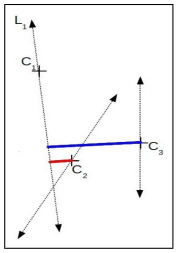


Figure 2. Shortest Distance

It can be noted that centers of the components from a same finger should lie on a line which is parallel to the dominant direction of the components. But often it is difficult to capture full part of a finger. So these centers are not exactly on that line but lie close to that line. One can see from Figure 1 that center of components shown with black square, D_8, D_9 and D_{10} lie close to the line L_1. Keeping this point in the mind, two components A and B are classified as belonging to the same finger if the shortest distance from center of the component B to the line formed by using center and orientation of component A is small. For example, in Figure 1, D_6 and D_7 should be merged as the shortest distance of the center of D_7 from the line L_2 formed by using center and orientation of D_6 is small. Notion of the shortest distance can be visualized using Figure 2 where centers and orientations of components C_1, C_2 and C_3 are shown by using + and dotted lines. Line L_1 passes through the center of C_1 and is oriented in its direction. Perpendicular distance of line L_1 from the center of C_2 is the shortest distance between C_1 and C_2. Algorithm 1 has been designed to calculate this shortest distance between components A and B, termed as $SD(A,B)$. As, the line L_1 is oriented in the same direction as that of component A so its slope m is given by

$$m = \tan(\theta_A) \tag{4}$$

where θ_A is the orientation of A. If (a_1, a_2) is the center of A, then the line with slope m and passing through the center satisfies

$$a_2 \;=\; m \times a_1 + c \tag{5}$$

The shortest distance between center of component B and the line L_1 can be found by

$$SD(A,B) = \frac{|b_2 - m \times b_1 \;-\; c|}{\sqrt{1+m^2}} \tag{6}$$

where (b_1, b_2) is the center of B and $c = \;a_2 \;-\; m \times a_1$ and m is the slope of L_1 Figure 2 shows that the shortest distance between C_1 and C_2, $SD(C_1, C_2)$, which is less than $SD(C_1, C_3)$. So C_2 and C_1 are more likely to be the members of the same finger.

Given the shortest distance between every pair of components Algorithm 2 determines different components of each finger and labels them. Let $Dist$ be a matrix which contains the shortest distance between each pair of components. The i^{th} row of $Dist$ provides the shortest distance of the component C_i from every other component. It is assumed that $Dist(i,i) = \infty$ (very large). Two components C_i and C_j are likely to be the member of the same finger if there does not exist any k such that

$$Dist(i,k) < Dist(i,j) \quad \forall k \neq j \tag{7}$$

Initially, a unique label is assigned to each component and is stored in an array CN.

If two components A and B are belonging to the same finger, then same component number has been assigned to A and B. So, this can be accomplished by following traditional Union-Find algorithm [1]. In order to avoid selection of same components again for merging, the shortest distance between these components is set to ∞. This process of merging and updation is iterated till each component belongs to one of the four fingers. Since it is assumed that slap image contains all four fingers, algorithm is iterated $n-4$ times, where n is the total number of components. One can observe that instead of global orientation of hand and local orientation of finger, only local orientation of the components has been used to merge the components which are belonging to a same finger.

Algorithm 1 $SD(A,B)$

Require: Component A with center (a_1, a_2) and orientation θ_A and, component B with center (b_1, b_2) and orientation θ_B.

Ensure: Return the distance of projection from point (b_1, b_2) to the line passing through (a_1, a_2) and oriented at θ_A, that is, $SD(A,B)$.

$m = \tan(\theta_A)$
$c = \; a_2 \;-\; m \times a_1$
$SD(A,B) = \frac{|b_2 - m \times b_1 \;-\; c|}{\sqrt{1+m^2}}$
return

3.2. Fingerprint Labeling

For each of these four fingers, components belong to a finger are treated as one entity. Orientations of components belonging to a finger is used to approximate the local orientation of the finger. Using the orientation and components belonging to a finger, one can determine the desired fingerprint of the finger. This approximation of orientation of a finger is much more accurate than the global orientation of the hand because it uses only those components which are present in that finger.

One can use euclidean distance between fingerprints to label as one of index, middle, ring and little finger. We have the following observations.

Algorithm 2 ComponentMerging $(C_1, C_2, ..., C_n)$

Require: n components with their $centers$ and $orientations.$

Ensure: Each component is given a number, such that, components belonging to same finger have same number.

for $i = 1$ to n **do**
  for $j = 1$ to n **do**
    if $i = j$ **then**
    $Dist(i, i) = \infty$
    else
    $Dist(i, j) = min(SD(C_i, C_j), SD(C_j, C_i))$
    end if
  end for
  $CN(i) = i$
end for
for $k = 1$ to $n - 4$ **do**
  Obtain(A, B) such that $Dist(A, B)$ is minimum in $Dist$
  Call $Union\text{-}Find$ algorithm to merge A and B and set $CN(A) = CN(B) = min(CN(A), CN(B))$
  Set $Dist(A, B) = Dist(B, A) = \infty$
end for
return

1. Euclidean distance between center of index finger and that of little finger is maximum.
2. Euclidean distance from index finger to the center of that of the middle finger is minimum, as compared to its distance to other fingers.
3. Euclidean distance of the center of little finger to that of the ring finger is minimum, as compared to its distance to other fingers.

Generally, little finger is smaller than index finger. Therefore, a hand type can be easily detected when end fingers are classified into little and index finger. This assumption along with the global orientation of the hand which can be found from these four components, can be used to detect the hand type. But global orientation estimation may not be always accurate as different fingers may have different orientation. In this paper, local orientations of these two fingers is used to find the relationship between neighboring fingers. It is a better estimate than that of estimating it using global orientation of hand because the differences of local orientation in neighboring fingers are low as compared to the differences of local orientation of the finger and global orientation of the hand. Finally, based on these relations between neighboring fingers, relationship between end fingers is established.

To determine, the relationship between a fingerprint and it's immediate right neighboring fingerprint, projection distance between them is used. Let $\vec{a}$ be a vector calculated by using the centers of fingerprints C^A and C^B. Then, $\vec{a}$ can be defined by

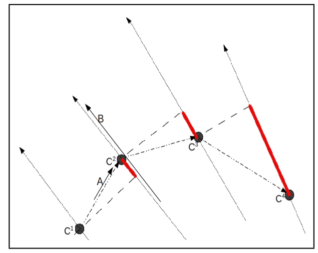


Figure 3. Projection Distance

$$\vec{a} = \left(x^A - x^B\right)\hat{i} + \left(y^A - y^B\right)\hat{j} \tag{8}$$

where $\left(x^A, y^A\right)$ and $\left(x^B, y^B\right)$ are the centers of C^A and C^B. Then, a vector $\vec{b}$ can be found out in the direction of C^B and passes through center of C^B as

$$\vec{b} = \left(x^B \times \cos\left(\Theta^B\right)\right)\hat{i} + \left(y^B \times \sin\left(\Theta^B\right)\right)\hat{j} \tag{9}$$

where Θ^B is the orientation of C^B. Based on these vectors, length of projection of $\vec{a}$ on $\vec{b}$ can be found by

$$D = \frac{a \cdot b}{|b|} \tag{10}$$

All the steps used to compute the projection distance are summarized in Algorithm 3. As an illustration, consider C^1 and C^2 shown in Figure 3, where $\vec{a}$ and $\vec{b}$ are shown using A and B. Projection distance between C^1 and its immediate right fingerprint component C^2 is marked in the figure and since A is in the direction of B, so it is positive in magnitude, representing that length of the finger C^2 is greater than that of finger C^1. Other projection distances are negative in magnitude, representing decrease in length of the finger going from left to right.

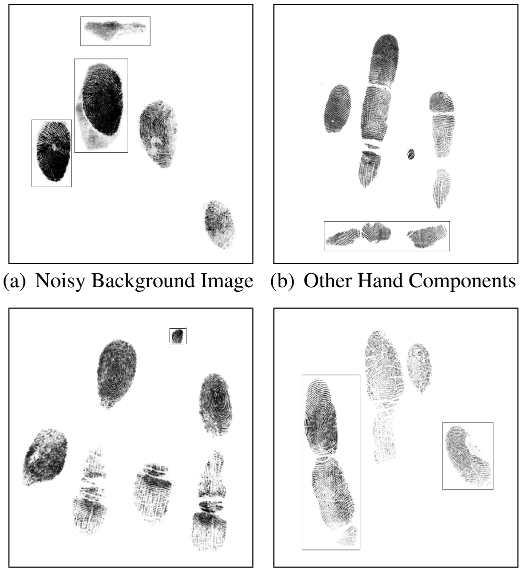
Algorithm 3 ProjectionDistance$\left(C^A, C^B\right)$

Require: Components C^A and C^B where C^A has centers $\left(x^A, y^A\right)$ with orientation Θ^A and C^B has centers $\left(x^B, y^B\right)$ with orientation Θ^B

Ensure: Distance of projection between C^A and C^B.

$\vec{a} = \left(x^A - x^B\right)\hat{i} + \left(y^A - y^B\right)\hat{j}$
$\vec{b} = \left(x^B \times \cos\left(\Theta^B\right)\right)\hat{i} + \left(y^B \times \sin\left(\Theta^B\right)\right)\hat{j}$
$D = \frac{a \cdot b}{|b|}$
return

In order to find the relationship between end fingers projection distances between all fingerprint components with there respective right neighboring fingerprint component are added. If this is negative, it implies that C^1 is the index finger and hence, slap-image consist of right hand as



(a) Noisy Background Image (b) Other Hand Components

(c) Small Fingertip Area (d) Wrongly Oriented Fingers

Figure 4. Causes of Failure

shown in Figure 3. Otherwise, slap-image is of left hand. Once a hand is detected, then each fingerprint component can easily be classified into their actual classes as index finger, middle finger, ring finger, or little finger of left hand or right hand.

4. Experimental Results

The proposed algorithm has been tested on a 6,096 slap images of 1,016 subjects. Each subject presents slap-images of left and right hand. Each image is of size 1500×1600 at 500ppi . These images are also tested for hand-type detection and the algorithm has made proper labeling of each finger. Components forming fingers for each slap-image are examined manually. Only 216 slap-images have shown spurious results due to following reasons

1. *Noisy background*: This is due to sweaty or latent fingerprints. This can result in merging two separate components or introducing a new component.

2. *Other hand components*: Sometime, at the time of enrollment thumb or other part of hand excluding fingers may touch the sensor, causing the presence of other components in slap image.

3. *Smaller size of fingertip component*: This is primarily due to small size of fingertip component. As a result, this component may be left out.

4. *Improper orientation*: Improper placement of hand can give rise to improper local orientations of the components in slap-image. This can lead to merging of inappropriate components.

All these factors can be visualized in Figure 4. The proposed algorithm performs successfully even when there is dull prints, large angle of rotation and non-elliptical shape of fingertips.

5. Conclusion

This paper has proposed an efficient algorithm to extract multiple fingerprints from a slap fingerprint image and to classify them into index, middle, ring or little finger of left or right hand. Entropy based segmentation has been used to detect the components even in the presence of large rotational angles, dull-prints and non-elliptical shape. Based on the geometrical and spatial constraints of a hand geometry, fingerprints are classified. The algorithm has accurately segmented each slap finger image into four fingerprints.

ACKNOWLEDGMENT

Authors are thankful to the anonymous reviewers for their valuable suggestions which have helped us to improve the quality of the manuscript. Authors also like to acknowledge the support provided by the Department of Information Technology, Government of India to carry out this work.

References

[1] A. Aho, J. Hopcroft, and J. Ullman. *Data structures and algorithms*. Addison-Wesley Longman Publishing Co., Inc., 1983.

[2] R. Hödl, S. Ram, H. Bischof, and J. Birchbauer. Slap fingerprint segmentation. 2009.

[3] P. Lo and P. Sankar. Slap print segmentation system and method, July 4 2006. US Patent 7,072,496.

[4] J. Pluim, J. Maintz, and M. Viergever. Mutual-information-based registration of medical images: a survey. *IEEE Transactions on Medical Imaging*, 22(8):986–1004, 2003.

[5] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1):71–86, 1991.

[6] B. Ulery, A. Hicklin, C. Watson, M. Indovina, and K. Kwong. Slap fingerprint segmentation evaluation 2004 slapseg04 analysis report. 2005.

[7] C. Wilson, R. Hicklin, H. Korves, B. Ulery, M. Zoepfl, M. Bone, P. Grother, R. Micheals, S. Otto, C. Watson, et al. *Fingerprint vendor technology evaluation 2003: Summary of results and analysis report*. US Dept. of Commerce, National Institute of Standards and Technology, 2004.

[8] Z. Yong-liang, L. Yan-miao, W. Hong-tao, H. Ya-ping, X. Gang, and G. Fei. Principal axis and crease detection for slap fingerprint segmentation. In *2010 17th IEEE International Conference on Image Processing (ICIP)*, pages 3081–3084. IEEE, 2010.

[9] Y. Zhang, G. Xiao, Y. Li, H. Wu, and Y. Huang. Slap fingerprint segmentation for live-scan devices and ten-print cards. In *20th International Conference on Pattern Recognition.(ICPR 2010)*, pages 23–26, 2010.