Available online at www.sciencedirect.com
SCIENCE DIRECT®
Pattern Recognition 39 (2006) 465–477
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/patcog

# Fingerprint matching by genetic algorithms

Xuejun Tan*, Bir Bhanu

*Center for Research in Intelligent System, University of California, Riverside, CA 92521, USA*

Received 24 February 2004; accepted 6 September 2005

**Abstract**

Fingerprint matching is still a challenging problem for reliable person authentication because of the complex distortions involved in two impressions of the same finger. In this paper, we propose a fingerprint-matching approach based on genetic algorithms (GA), which tries to find the optimal transformation between two different fingerprints. In order to deal with low-quality fingerprint images, which introduce significant occlusion and clutter of minutiae features, we design a fitness function based on the local properties of each triplet of minutiae. The experimental results on National Institute of Standards and Technology fingerprint database, NIST-4, not only show that the proposed approach can achieve good performance even when a large portion of fingerprints in the database are of poor quality, but also show that the proposed approach is better than another approach, which is based on mean-squared error estimation.


*Keywords:* Fitness value; Corresponding triangles; Minutiae; Optimization; Fingerprint verification

## 1. Introduction

Fingerprint matching is one of the most promising methods among biometric recognition techniques and has been used for person authentication for a long time. Now, it is not only used by police for law enforcement, but also in commercial applications, such as access control and financial transactions. In terms of applications, there are two kinds of fingerprint recognition systems: verification and identification. In verification, the input is a query fingerprint and an identity (ID), the system verifies whether the ID is consistent with the fingerprint. The output is an answer of yes or no. In identification, the input is only a query fingerprint, the system tries to answer the question: Are there any fingerprints in the database that resemble the query fingerprint? The output is a short list of fingerprints. In this paper, we are dealing with the verification problem.

A fingerprint is formed by a group of curves. The most useful features, which include endpoints and bifurcations, are called minutiae. Fig. 1 shows examples of an endpoint and a bifurcation in a fingerprint image. In previous work (e.g. [30]), we present a learned templates-based algorithm for minutiae extraction. Templates are learned from examples by optimizing a criterion function using Lagrange's method. To detect the presence of minutiae in fingerprints, templates are applied with appropriate orientation to the binary fingerprints only at selected potential minutia locations.

Generally, the minutiae-based fingerprint verification is a kind of point-matching algorithm. However, the distortions between two sets of minutiae extracted from the different impressions of the same finger may include significant translation, rotation, scale, shear, local perturbation, occlusion and clutter, which make it difficult to find the corresponding minutiae reliably.

## 2. Related work and contribution

### 2.1. Related work

Generally, fingerprint-matching algorithms have two steps: (1) align the fingerprints and (2) find the correspondences between two fingerprints. The approach proposed

* Corresponding author.
*E-mail addresses:* xtan@cris.ucr.edu (X. Tan), bhanu@cris.ucr.edu (B. Bhanu).

0031-3203/$30.00 
doi:10.1016/j.patcog.2005.09.005

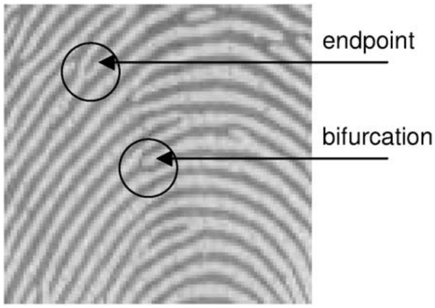


Fig. 1. Examples of minutiae.

by Jain et al. [2] is capable of compensating for some of the nonlinear deformations and finding the correspondences. However, since the ridges associated with the minutiae are used to estimate the alignment parameters, the size of the templates has to be large, which takes much memory and computation, otherwise, the alignment will be inaccurate. Jiang and Yau [3] use the local and global structures of minutiae in their approach. The local structure of a minutia describes a rotation and translation invariant feature of the minutia in its neighborhood, and the global structure tries to determine the uniqueness of a fingerprint. The problem with this technique is that it cannot compensate for real-world distortions of a 3D elastic finger. These distortions can be considered equivalent to a space variant scale distortion. Furthermore, the weight vector that is associated with each component of the feature vector, such as distances, directions, relative local orientations, etc., has to be empirically determined. Another prominent matching algorithm, which is proposed by Kovacs-Vajna [4], uses triangular matching to deal with the deformations of fingerprints. However, the final results of matching have to be validated by a dynamic time warping (DTW) algorithm. Without DTW for further verification, the results are not acceptable. In previous work (e.g. [5]), we have developed a fingerprint identification approach, which is based on the local optimization of the corresponding triangles to perform verification between two fingerprints.

Besides minutiae, researchers have also used other features for fingerprint matching. Saleh and Adhami [6] proposed an approach which transforms fingerprint images into a sequence of points in the angle-curvature domain. The matching between a query fingerprint and a template fingerprint is based on the least-squares error of the Euclidean distance between corresponding points in the angle-curve domain. Jain et al. [7] presented a filter-based algorithm, which uses a bank of Gabor filters to capture both local and global details in a fingerprint as a compact fixed length FingerCode. The authors reported that the FingerCode-based system performs better than a state-of-the-art minutiae-based system when the performance requirement of the application system does not demand a very low false acceptance rate.

The combinations of different kinds of features have also been used in fingerprint matching. Jain et al. [8] presented a hybrid-matching algorithm that uses both minutiae and texture information. Ceguerra and Koprinska [9] proposed an approach that uses matched minutiae as the reference axis to generate a shape signature for each fingerprint. The shape signature is then used to form a feature vector describing the fingerprint. A linear vector quantizer (LVQ) neural network is trained using the feature vectors to match fingerprints. Both approaches reported improvements in the matching results.

### *2.2. Contribution*

In this paper, we use genetic algorithms (GA) to achieve a globally optimized solution for the transformation between two sets of minutiae extracted from two different fingerprints. The fitness function is based on the local properties of each triplet of minutiae, which include angles, triangle handedness, triangle direction, maximum side, minutiae density and ridges counts. The performance of our approach on the NIST-4 database, which has a large portion of fingerprints of poor quality, shows that our approach can tolerate highly nonlinear deformations. The comparison of the proposed approach with another approach based on mean-squared error estimation shows the advantage of GA-based verification.

## 3. Technical approach

In Ref. [10] we have provided an analysis of features (see Section 3.3.2), specifically the angles that are used as features. By using these features we are able to handle complex distortions encountered in fingerprint images. Thus, we can use a simple transformation consisting of scale, rotation and translation for matching between a template and a query fingerprint in this paper.

### *3.1. Fingerprint-matching problem*

Suppose the sets of minutiae in the template and the query fingerprints are $\{(x_{n,1}, x_{n,2})\}$ and $\{(y_{m,1}, y_{m,2})\}$, respectively, where $n = 1, 2, 3, \ldots, N$, $m = 1, 2, 3, \ldots, M$, $(x_{n,1}, x_{n,2})$ and $(y_{m,1}, y_{m,2})$ are the coordinates of minutiae. The number of minutiae in the template and the query fingerprints are $N$ and $M$, respectively. The transformation $Y_i = F(X_i)$ between $X_i(x_{i,1}, x_{i,2})$ and $Y_i(y_{i,1}, y_{i,2})$ can be simplified as

$$Y_i = s \cdot R \cdot X_i + T \tag{1}$$

where $s$ is the scale factor,

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix},$$