

A TSVM-based Minutiae Matching Approach for Fingerprint Verification

Jia Jia, Lianhong Cai

Key Laboratory of Pervasive Computing (Tsinghua University), Ministry of Education
Beijing 100084, P.R.China
{jiajia}@mails.tsinghua.edu.cn
{clh-dcs}@tsinghua.edu.cn

Abstract. This paper introduces Transductive Support Vector Machine (TSVM) into fingerprint verification. An improved fingerprint matching approach using TSVM is presented. In the proposed approach, the traditional minutiae-based fingerprint matching task is transformed to a classification task using TSVM. The paper presents an analysis of why TSVM are well suited for fingerprint matching, especially for small training sets. The approach is supported by experiments on five test collections, including both international and domestic fingerprint verification competition databases. Experimental results show that our approach is insensitive to noise as well as with effective performance.

1 Introduction

Fingerprint authentication is one of the most important biometric technologies [1]. Fingerprint is the pattern of ridges and valleys on the surface of fingers. As the fingerprint of a person is unique and immutable, automatic fingerprint verification system (AFVS) can be widely used in both anti-criminal and civilian applications. Therefore, performance improvements are the key points in current research on AFVS.

Operationally, AFVS can be decomposed into two steps: fingerprint preprocessing and fingerprint matching. Our main work focuses on the minutiae-based matching approach. Minutiae are the pattern of ridge bifurcations and endings. They are extracted from the thinned image which is obtained in the preprocessing step [2, 5, 6], as shown in Fig.1.



Fig. 1. Fingerprint minutiae. A ridge ending and a ridge bifurcation

Traditional minutiae-based matching algorithms (TMMA) use minutiae localizations (positions and orientations) [3], or combine them with texture information [4, 6] as the fingerprint features. After getting the feature vector which is the matching result of the query and template images' features, TMMA normally determine the final conclusion, fingerprints matched or not, by comparing a matching score with a *certain threshold* (Fig.2). The matching score is the dot product of the feature vector and a *certain weight vector*. The higher the matching score the better the match.

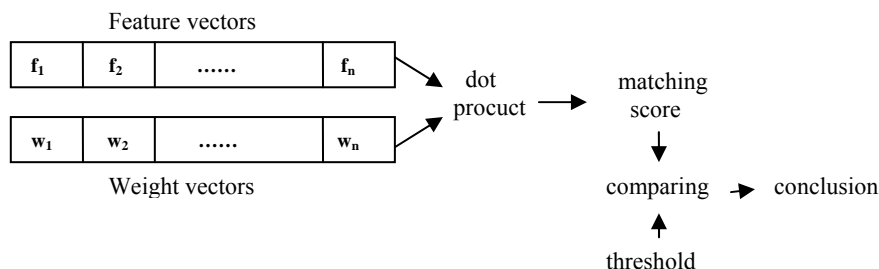


Fig. 2. The final conclusion is determined by comparing the matching score with a certain threshold. The matching score = $(f_1, f_2, \dots, f_n) \cdot (w_1, w_2, \dots, w_n)$, where f_n is a feature dimension and w_n is the corresponding weight dimension

So our proposed approach is motivated by the following observations:

- 1) Due to the various skin and impression conditions, *the certain weight vector* may lead a false conclusion to the final result. For example, we define a weight vector $w = (0.5, 0.5)$ and a feature vector $f = (f_1, f_2)$ for a fingerprint database, where f_1 is the total number of matched minutiae and f_2 presents the similarity of two fingerprints' texture. In such case, if the sensors capture only a small area of the fingertip, which means a query and a template image have a small overlap of common region (Fig.3), the matching score = $0.5f_1 + 0.5f_2$ may have a low value because of the low value of f_1 . Finally the low matching score will lead to a "not matched" result although the two images have a high similarity of texture.



Fig. 3. Example of fingerprint images that having small overlap of common region but high similarity of texture. The two fingerprints is captured from the some finger

- 2) *The certain threshold* for result determination should be evaluated according to the different fingerprint images' quality for different databases. This course is time-consuming, and may be inaccurate to the final conclusion (Fig.4(a)).

In order to solve these two problems, the appropriate way is to avoid using certain weights and certain threshold, but divide the feature vectors standing for matched or not by a machine learning technique. Our key insight is that the matching problem can be regarded as a two-class classification problem (samples matched or not), so some classification methods, such as the multidimensional classifier using a hyper-plane, can be adopted. That means we transform the problem which is hard to be solved in one-dimension space to a multidimensional space. This can be explained clearly by Fig.4(b).

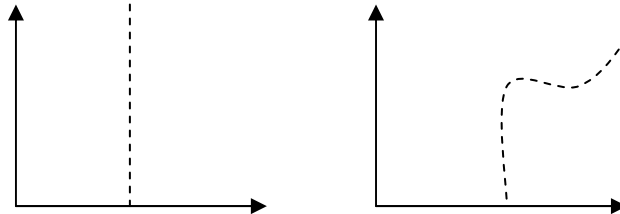


Fig. 4. White circles standing for the positive samples and the black ones standing for the negatives. (a) For the positive and negative samples, a certain one-dimensional threshold can not divide them exactly; (b) For the positive and negative samples, their projection to either axis has superposition, but in two-dimensional space, they are linear-dividable

Therefore, we propose a new fingerprint matching approach that transforming the matching problem to a classification problem and choosing a classifier TSVM [10] to solve it. The approach includes the following steps: 1) extracting and purifying the minutiae localizations to get the real minutiae information; 2) using minutiae localizations for the global matching to get the total number of matched minutiae; 3) defining a matching vector standing for the similarity of two fingerprints according to the results of 2); 4) using TSVM to label the matching vector with conclusion "matched or not". The performance of our approach is illustrated over both international and domestic fingerprint verification competition datasets, and experimental results show substantial improvements in the performance of fingerprint verification.

The rest of this paper is organized as follows. In section 2, we describe the minutiae extraction and purification scheme. In section 3, our matching approach based on TSVM is presented in detail, and we also contrast TSVM with general SVM to give the reasons of choosing TSVM as the classifier. Before concluded in section 5, experimental results and discussions are given in section 4.

2 Minutiae Extraction and Purification

Before matching step, we first present a minutiae extraction and purification scheme which is based on binary and thinned fingerprint images to obtain real minutiae [2].

For each minutia, we define a 4-tuple $(x, y, type, theta)$ to describe its features. The x and y indicate the minutia's coordinate. The value of $type$, 1 or 2, indicates that the minutia is a ridge ending or a bifurcation. The $theta$ indicates the minutia's orientation. Here we use an important concept, used in Xiao and Raafat [11], which will help us find the endings and bifurcations in the thinned image is the *condition number* (C_N).

Definition 1. The condition number of a given foreground pixel is:

$$C_N = \sum_{k=1}^8 |\Gamma(k+1) - \Gamma(k)| \text{ where } \Gamma(9) = \Gamma(1) \quad \Gamma(p) = \begin{cases} 1, & \text{if pixel } p \text{ is foreground;} \\ 0, & \text{otherwise;} \end{cases} \quad (1)$$

the index k scans the eight neighbors of the pixel in clockwise direction.

If C_N is 2, the pixel is an ending and, if C_N is 6, the pixel is a bifurcation. All other values of C_N are ignored.

Definition 2. A minutia's orientation $theta$ is the local ridge orientation of the associated ridge.

Divide the fingerprint image into blocks of size $W \times W$. Compute the gradients G_x and G_y at each pixel in each block. Then the local orientation of each block is [5]

$$\theta(i, j) = \frac{1}{2} \tan^{-1} \frac{\sum_{u=i-w/2}^{i+w/2} \sum_{v=j-w/2}^{j+w/2} 2 G_x(u, v) G_y(u, v)}{\sum_{u=i-w/2}^{i+w/2} \sum_{v=j-w/2}^{j+w/2} (G_x^2(u, v) - G_y^2(u, v))} \quad (2)$$

Due to variations in skin conditions like postnatal marks or occupational marks, and impression conditions, a significant percentage of fingerprint images are in poor quality, in which ridge structures are corrupted. This leads to the problem that a large amount of false minutiae may be introduced. So it's clear that a post-processing stage is necessary to purify minutiae before matching.

The false minutiae include small holes, breaks, bridges, and spurs and so on. Here we choose a minutiae purify algorithm using local ridge structure [12]. The purification course is shown in Fig 5.

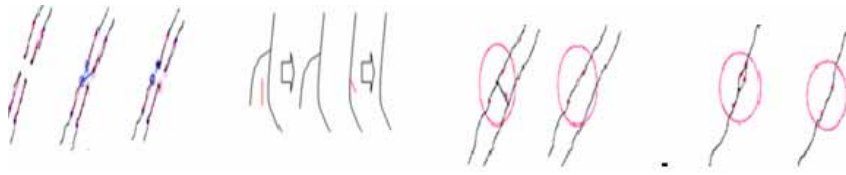


Fig. 5. Experimental results of minutiae purification: breaks, spurs, bridges and holes

3 A Matching Approach Based on TSVM

3.1 SVM VS. TSVM^[13]

Support Vector Machine [7, 8] is a powerful classification method. It is based on a solid theoretical foundation – *structural risk minimization* [9]. In its simplest linear form, an SVM is a hyperplane that separates the positive and negative training examples with maximum margin, as shown in Fig 6. Large margin between positive and negative examples has been proven to lead to good generalization [9].

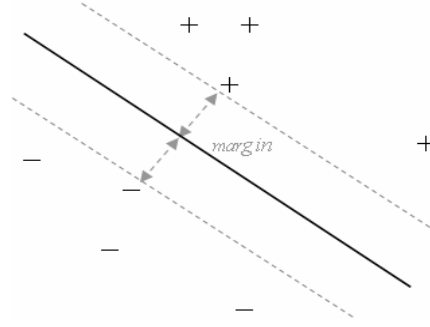


Fig. 6. An SVM/TSVM is a hyperplane that separates the positive and negative examples with maximum margin. The examples closest to the hyperplane are called support vectors

Transductive SVM (TSVM) introduced by Joachims [10] extends SVM to transductive learning setting. A TSVM is essentially a hyperplane that separates the positive and negative training examples with maximum margin on both training and test examples.

A regular SVM tries to induce a general classifying function which has high accuracy on the whole distribution of examples. However, this so-called inductive learning setting is often unnecessarily complex [13]. In fact, we do not care about the general classifying function, but rather attempt to achieve good classification performance on set of test examples. This is exactly the goal of transductive learning [9].

Why can TSVM be better than SVM or other classifier for fingerprint verification? There usually exists a clustering structure of training and test examples in fingerprint datasets: the examples in same class, which means from the same finger, tend to be close to each other in feature space. As explained in [10], it is this clustering structure of examples that TSVM exploits as prior knowledge to boost classification performance. This is especially beneficial when the number of training examples is small, like for fingerprint verification competition training case.

3.2 Global Matching

Global matching is the step to compute total number of matched minutiae between the test image and template image.

To ensure an overlap of common regions, the two images need to be aligned first. This is done by determining the transformation parameters (t_x, t_y, ρ, θ) , where t_x, t_y indicate the adjustable distances in x-axis and y-axis, ρ indicates the flex coefficient and θ indicates the rotated angle. (t_x, t_y, ρ, θ) is computed by coordinates of two pairs of minutiae (usually delta points and core points) from both images [3]. These two pairs of minutiae are called reference points.

Once the transformation parameters (t_x, t_y, ρ, θ) are obtained, the test image can be aligned. Let (x, y) represent the original coordinate, then the aligned coordinate (u, v) , is obtained as Eq.3.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \rho \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (3)$$

Due to the structures of fingerprints themselves and the conditions of impression, some fingerprint images do not have delta points or center points [5]. In such cases, every two pairs of minutiae from test image and template image, if the minutiae of pair having the same value of *type*, should be chosen to be as reference points, and get the corresponding transformation parameters (t_x, t_y, ρ, θ) . For different reference points, there will be different numbers of matched minutiae, and the maximum number indicates the similarity of the two images.

The pseudo code of our global matching scheme is given below:

```

Algorithm global matching
Objective: compute the total number of matched minutiae
between the test image and template image
Input:
   $I_{test}$  : the test image.
   $I_{temp}$  : the template image.
   $\{p_n\}$  : the minutiae sequence of  $I_{test}$ .
   $\{q_m\}$  : the minutiae sequence of  $I_{temp}$ .
Output:
  the maximum number of matched minutiae pairs:  $t$ .
Method:
1. For  $p_i, q_j$  ( $1 \leq i \leq n, 1 \leq j \leq m$ )
   if  $type[p_i] = type[q_j]$  add  $(p_i, q_j)$  to set  $\mathbf{A}$ 
Forall  $(p_{i1}, q_{j1}), (p_{i2}, q_{j2}) \in \mathbf{A}$ , do Step2 to Step5
2. compute  $(t_x, t_y, \rho, \theta)$  by Eq. (1)
3. select  $(t_x, t_y, \rho, \theta)$ 
   if  $|\rho - 1| > 0.1$  or  $\theta > \pi/3$  skip Step4 and Step5
//This selection ensures that the flex coefficient  $\rho$  ap-
proximates to 1 and the rotation angle  $\theta$  is less than  $60^\circ$ 
4. For  $\{p_n\}$ 
   compute  $(x_i, y_i, theta_i)_{new}$  by Eq. (1) to gain  $\{p_n\}_{new}$ 
5. match  $\{p_n\}_{new}$  with  $\{q_m\}$ . Record the number of matched minu-
tiaie pairs. Go Step2
6. return  $t$  and the corresponding  $(t_x, t_y, \rho, \theta)$ 

```

Fig. 7. Pseudo code for global matching scheme

3.3 Getting the Matching Vector

We define a matching vector $\mathbf{V}(n, m, t, err)$ to describe the similarity of I_{test} and I_{temp} . This vector \mathbf{V} is obtained by the next two steps:

Step 1: Record the minutiae number n of I_{test} and m of I_{temp} , and maximum matched minutiae number t .

Step 2: Get a $100\text{pixel} \times 100\text{pixel}$ sub-image I_{sub} from the center of the threshold image of I_{test} . Translate I_{sub} by parameters (t_x, t_y, ρ, θ) to I_{sub}' , and let err represent the number of pixels in I_{sub}' that have the same intensity as its corresponding pixel in I_{temp} .

The err describes an estimation of the matching error.

3.4 Determining the Results by Matching Vector and SVM/TSVM

For a matching vector $\mathbf{V}(n, m, t, err)$, we need to label it with “matching success” or “matching failure”. The decision function of an SVM/TSVM is shown in Eq.4.

$$f(\mathbf{V}) = \langle \mathbf{w} \cdot \mathbf{V} \rangle + b \quad (4)$$

$\langle \mathbf{w} \cdot \mathbf{V} \rangle$ is the dot product between \mathbf{w} (the normal vector to the hyperplane) and \mathbf{V} (the matching vector). The margin for an input vector \mathbf{V}_i is $y_i f(\mathbf{V}_i)$ where $y_i \in \{-1, 1\}$ is the correct class label for \mathbf{V}_i . Seeking the maximum margin can be expressed as minimizing $\langle \mathbf{w} \cdot \mathbf{w} \rangle$ subject to $y_i (\langle \mathbf{w} \cdot \mathbf{V}_i \rangle + b) \geq 1, \forall i$. We allow but penalize the examples falling to the wrong side of the hyperplane.

4 Experiments and Discussion

We conducted experiments with data of fingerprint verification competitions, to demonstrate the advantages of our proposed approach to fingerprint verification.

4.1 Datasets

In order to prove the influence of different image qualities and image amount to our matching approach, we have collected 5 datasets from FVC2002¹ (The Second International Fingerprint Verification Competition) and BVC2004² (Biometrics Verification Competition 2004). The information of each dataset is shown in Table 1.

Each fingerprint image allows a rotation angle that belongs to $[-\pi/4, \pi/4]$ (compared with the vertical line). Every two images from one finger have an overlap of common region. But there may be no delta points or core points in some fingerprint images.

Table 1. The information of datasets

	The source of the datasets	different fingers / total images	Sensors	Image size	Resolution
1 st db	BVC2004 DB1	40/400	Optical sensor	412 x 362	500 dpi
2 nd db	BVC2004 DB2	40/400	CMOS sensor	256 x 300	500 dpi
3 rd db	BVC2004 DB3	40/400	Thermal sweeping sensor	300 x 480	500 dpi
4 th db	BVC2004 DB4	40/400	Fingerpass	380 x 460	500 dpi
5 th db	FVC2002 DB1	230/1840	Optical sensor	388 x 374	500 dpi

4.2 Experiments Setup

We posed 2 experiments. For each experiment, we compared our approach using TSVM with SVM and also with TMMA which uses threshold for result determination [5]. All approaches use the same image preprocessing algorithms [2]. Both the experiments are done by the method of 5-folder cross validation, but have differences in the size of test sets and training sets.

Experiment 1. For db1 to db4, 400 images are divided into 5 parts, each of which has 80 images. Both the algorithms run five times. For each time, **four** of the five parts are used as training sets (our approach only), and the other **one** part is used as test set. The averaged verification result will be reported over these 5 times.

Experiment 2. For db5, 1840 images are divided into 5 parts, each of which has 368 images. Both the algorithms run five times. For each time, **one** of the five parts is used as training set (our approach only), and the other **four** parts are used as test sets. The averaged verification result will be reported over these 5 times.

We use SVMlight³ for the implementation of SVM/TSVM [8, 10], and take linear kernel in experiments.

4.3 Measures

The performance of a fingerprint verification algorithm can be measured by FNMR (False Non Match Rate: each sample in the subset A is matched against the remaining samples of the same finger), FMR (False Match Rate: the first sample of each finger in the subset A is matched against the first sample of the remaining fingers in A) and the average time of matching two images. Especially for Experiment 2, we also measure the maximum memory size of our approach. The configuration of running computer is PIV1.0G, 256M DDR.

¹ <http://bias.csr.unibo.it/fvc2002/>

² <http://www.sinobiometrics.com/chinese/conferences/sinobiometrics%2704.htm>

³ <http://svmlight.joachims.org/>

4.4 Results and Discussions

Table 2. Experimental results of db1 to db4

	Avg match time (TSVM)	FMR	FNMR		
			TMMA	SVM	TSVM
1 st database	0.901 sec	~ 0	6.67%	2.03%	1.94%
2 nd database	0.524 sec	~ 0	30.28%	12.78%	9.31%
3 rd database	1.004 sec	~ 0	24.17%	7.78%	7.50%
4 th database	0.933 sec	~ 0	13.05%	5.28%	5.14%

Table 3. Experimental results of db5

FNMR			FMR	Avg match time (TSVM)	Max match Mem (TSVM)
TMMA	SVM	TSVM			
13.84%	4.08%	4.03%	~ 0.1%	0.459sec	8884 Kbytes

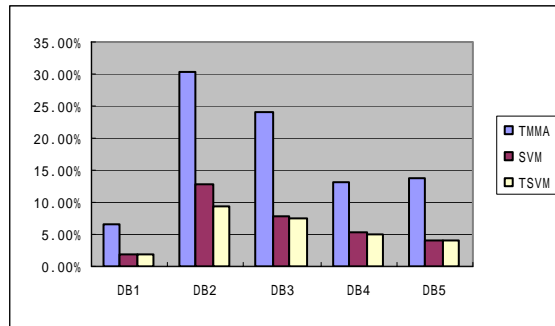


Fig. 8. Comparing the FNMR of TMMA with our approach using SVM and TSVM in five different datasets

The experimental results of db1 to db4 are shown in Table 2 and the results of db5 are shown in Table 3. We see that our approach using SVM/TSVM really can achieve much better accuracy than TMMA for fingerprint verification, and the average time the maximum memory for matching a pair of fingerprints is also acceptable.

As shown in Table 1, fingerprints of these five datasets are captured by sensors of different types. So the images have different qualities. This strongly suggests that our feature extraction and purification scheme and SVM/TSVM methods capture well the information needed for fingerprint verification, and have a low influence by fingerprint image quality.

We see that although the proportion of training sets is reduced, and the number of test members is increased in db5, our approach using SVM/TSVM still works better than TMMA. This implies that these machine learning methods have a low influence by fingerprint image amount.

Comparing the experimental results of SVM and TSVM, it turns out that the transductive learning technique can provide some help to fingerprint matching. We think this is because TSVM makes effective use of the matching vectors to enhance classification.

The experimental results of our approach and TMMA are compared in Figure 5. It is clear that our matching approach outperforms TMMA consistently and significantly.

5 Conclusion

Our main contributions to fingerprint verification are: 1) introduces SVM /TSVM for fingerprint verification that transforming the traditional minutiae-based matching task

to a classification task, and using a powerful classifier to solve it; 2) proposing a corresponding definition of matching vectors which capture well the information needed for SVM/TSVM classifier; 3)proposing a corresponding minutiae extraction and purification scheme to getting minutiae information for fingerprint images of poor quality.

Future work may include: looking for a more accurate fingerprint preprocessing a method to accelerate the proposed TSVM approach, and so forth.

Acknowledgment

I would like to thank reviewers for their kindly review and helpful comments for this paper. ☺

References

1. Newham, E.: The Biometric Report. SJB Services. New York (1995)
2. Lin, H., Wang, Y.F., Anil, J.: Fingerprint Image Enhancement: Algorithm and Performance Evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, No.8 (1998)
3. Wang, C.W., Li J.W.: Fingerprint Identification Using Point Pattern Matching. *Journal of Chongqing University (Natural Science Edition)*, Vol. 25, No.6 (2002)
4. Ross, A., Anil, J.: A Hybrid Fingerprint Matcher. *Pattern Recognition*, 36 (2003) 1661-1673
5. Anil, J., Lin, H.: On-line fingerprint verification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No.4 (1997) 302-314
6. Anil, J., Ross A., Prabhakar S.: Fingerprint Matching using Minutiae and Texture Features. In: *International Conference on Image Processing (ICIP)*. Thessaloniki, Greece (2001) 282-285
7. Cristianini, N., Shawe-Taylor J.: *An Introduction to Support Vector Machines*. Cambridge University Press. Cambridge, UK (2000)
8. Yuan, Y., Paolo, F., Massimiliano P.: Fingerprint Classification with Combinations of Support Vector Machines. *Lecture Notes in Computer Science*, Vol. 2091. Springer-Verlag, Berlin Heidelberg New York (2001) 253–258
9. Vapnik, V.N.: *The Nature of Statistical Learning Theory*. Springer-Verlag, Berlin Heidelberg New York (2000)
10. Joachims, T.: Transductive Inference for Text Classification using Support Vector Machines. In: *Proceedings of the 16th International Conference on Machine Learning (ICML)*. Bled, Slovenia (1999) 200-209
11. Xiao, Q., Raafat, H.: *Pattern Recognition*, 24,10 (1991) 985-992
12. Kasaei, S., Deriche, M.: Fingerprint Feature Extraction Using Block-Direction on Reconstructed Images. *Journal of Speech and Image Technologies for Computing and Telecommunications*, IEEE Tencon (1997)
13. Zhang, D., Lee, W.S.: Web Taxonomy Integration using Support Vector Machines. In: *Proceedings of WWW 2004*, New York, USA. ACM 1-58113-844-X/04/0005