



<b>Title</b>	<b>Improvement of fingerprint retrieval by a statistical classifier</b>
<b>Author(s)</b>	<b>Leung, KC; Leung, CH</b>
<b>Citation</b>	<b>IEEE Transactions On Information Forensics And Security, 2011, v. 6 n. 1, p. 59-69</b>
<b>Issued Date</b>	<b>2011</b>
<b>URL</b>	<b><a href="http://hdl.handle.net/10722/139266">http://hdl.handle.net/10722/139266</a></b>
<b>Rights</b>	<b>IEEE Transactions on Information Forensics and Security. Copyright © IEEE.</b>

# Improvement of Fingerprint Retrieval by a Statistical Classifier

K. C. Leung and C. H. Leung

**Abstract**—The topics of fingerprint classification, indexing, and retrieval have been studied extensively in the past decades. One problem faced by researchers is that in all publicly available fingerprint databases, only a few fingerprint samples from each individual are available for training and testing, making it inappropriate to use sophisticated statistical methods for recognition. Hence most of the previous works resorted to simple  $k$ -nearest neighbor ( $k$ -NN) classification. However, the  $k$ -NN classifier has the drawbacks of being comparatively slow and less accurate. In this paper, we tackle this problem by first artificially expanding the set of training samples using our previously proposed spatial modeling technique. With the expanded training set, we are then able to employ a more sophisticated classifier such as the Bayes classifier for recognition. We apply the proposed method to the problem of one-to- $N$  fingerprint identification and retrieval. The accuracy and speed are evaluated using the benchmarking FVC 2000, FVC 2002, and NIST-4 databases, and satisfactory retrieval performance is achieved.

**Index Terms**—Distorted sample, fingerprint identification, fingerprint retrieval, FVC database, NIST-4 database.

## I. INTRODUCTION

**F**INGERPRINT is the most commonly used biometric property in security, commerce, industrial, civilian, and forensic applications [1]. Over the years, researchers have proposed different problem formulations to be tackled by their systems. Those systems can be clustered into three major categories, namely, fingerprint classification, verification, and indexing/retrieval [1].

Fingerprint classification refers to the problem of assigning a given fingerprint into a predefined class (e.g., Henry class) based on its global structure and singular points [2]–[4]. A fingerprint verification system authenticates a person's identity by comparing the captured fingerprint with his/her own previously enrolled reference template stored in the database [5], [6]. It conducts a one-to-one comparison to confirm whether or not the claim of identity by the individual is true.

In fingerprint indexing/retrieval, the problem of one-to-one matching is extended to one-to- $N$  matching without requiring the subject's claim of identity. Given a fingerprint instance of

unknown identity, the system searches through the entire database of enrolled templates and returns a list of probable fingers (identifiers of individual) that the fingerprint may belong to [1]. The fingerprint features chosen to train the system should be reliable enough to distinguish one finger from all the others in the database. To this aim, some reported works made use of the orientation field features while some explored the features on the complex-filtered images. Minutiae features and other transformed features were also adopted. Details will be given in Section II.

One common problem in pattern recognition is the lack of samples in training a classifier. The curse of dimensionality often prohibits classifiers to be sufficiently trained, especially in high dimensional space [7]. Unfortunately, the number of samples per finger that is available for recognition is rather limited. Many publications about fingerprint retrieval resorted to  $k$ -nearest neighbor ( $k$ -NN) classification, in which  $k$  was often set to 1 and the distance metric was often Euclidean. The Fingerprint Verification Competition (FVC) databases (2000, 2002, and 2004) contain only eight instances per finger in each set [8]–[10]. Even worse, the National Institute of Standards and Technology database 4 (NIST-4) contains only two instances per finger [11]. This poses a big challenge to researchers trying to solve the problem of fingerprint retrieval or identification.

We have attempted to tackle the problem of insufficient training samples by generating additional samples, emulating the genuine samples that would have been captured by the fingerprint sensing device during fingerprint enrollment [12]. In our previous work, we have tried to produce more training samples for the NIST-4 database by three kinds of spatial modeling, namely, translational modeling, rotational modeling, and distorted sample generation. The distortion model employed in our work has been constructed based on the publications on plastic deformation of fingerprint images [13]–[15]. We have concluded that the Euclidean 1-NN retrieval performance has been greatly improved with the help of these artificial samples. At the same time, of course, the classification speed and required storage have been compromised.

The objective of this research is to build a fingerprint retrieval system that is accurate and efficient, with the goal to raise the recognition rate for the first few top rank candidates. This is achieved by extending our previous work in which additional training samples are generated by spatial modeling. With a lot more training samples, it becomes possible to train up a more sophisticated classifier instead of the 1-NN classifier. In this work, we adopt the Bayes classifier with the assumption of Gaussian statistics.

Extensive simulations with the FVC 2000, FVC 2002, and NIST-4 databases have confirmed the usefulness of our pro-

Manuscript received June 07, 2010; revised November 22, 2010; accepted November 29, 2010. Date of publication December 17, 2010; date of current version February 16, 2011. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Fabio Scotti.

The authors are with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong (e-mail: kachung@eee.hku.hk; chleung@eee.hku.hk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TIFS.2010.2100382

posed approach. The first rank identification error has been reduced by about 30% to 60%, compared with 1-NN classification without additional samples. In most of the time, our system offers comparable performance among other published works with similar goal and database under test. The gain in computational efficiency of Bayes classification (assuming Gaussian statistics) is also significant when more samples are available.

In Section II, we present a review on the state-of-the-art retrieval methods. Section III summarizes the baseline retrieval system using the Euclidean 1-NN classification. In Section IV, we describe our proposal to incorporate Bayes classification on top of the baseline system to improve the recognition performance. In Section V, we present our simulation results and compare them with those in the literature, followed by timing analysis and discussions. We then conclude our paper in Section VI.

## II. LITERATURE REVIEW

An early attempt of fingerprint retrieval based on Euclidean distance measure was done by Lumini *et al.* [16]. By using orientation field vectors as features, they concluded that their approach of retrieval had directed to a much better retrieval performance on the NIST-4 database than that of the five-class classification applications. Cappelli and Lumini [17] then extended the work by using a structural method called dynamic image partitioning. They utilized cost functions to quantify the effort of applying different types of dynamic mask and hence the likelihood measurement. Improved retrieval performance was obtained, especially in the range of high penetration rates.

Another work with respectable retrieval performance on the NIST-4 database based on minutiae triplets was done by Bhanu and Tan [18]. Later on, Li *et al.* [19] and Liu *et al.* [20] proposed the use of complex filter responses in feature vectors for accurate retrieval. Jiang *et al.* [21] and Liu *et al.* [22] then refined the distance metrics and incorporated an auxiliary feature (dominant ridge distance) to greatly boost the retrieval performance. Recently, Gyaourova and Ross [23] invented a new index coding scheme by employing reference images and evaluated their system on the same database.

Fingerprint indexing/retrieval was also conducted on the FVC databases, usually on the FVC 2000 Db2, FVC 2000 Db3, and FVC 2002 Db1. Cappelli *et al.* [24] first reported the retrieval performance on the FVC 2000 Db2 by orientation field features with multidimensional K-L transform. Jiang *et al.* [21] achieved improved figures on the same database by their proposed weighted distance measure and auxiliary feature, lowering the error rates drastically. Besides them, De Boer *et al.* [25] combined various fingerprint features and reported the retrieval performance on the FVC 2002 Db1.

While minutiae features were found to be very helpful in fingerprint matching, Liang *et al.* [26], [27] showed that they had also led to remarkable retrieval performance on the FVC databases. Recently, Feng and Cai [28] and Shuai *et al.* [29] implemented ridge invariant features and scale invariant feature transformation, respectively, for improving the indexing performance on the FVC 2002 Db1. Another work that explicitly formulated a fingerprint identification task and analyzed the first rank recognition accuracy attributes to Sujan and Mulqueen [30]. However, only 15 fingers were used in their experiment.

Due to the relatively low reliability of fingerprint retrieval, some previous works put emphasis on the overall retrieval performance at certain penetration rates of the database (e.g., from 5% to 30%), omitting the recognition rates of the top rank candidates [16], [17], [20]–[23]. However, it is more interesting to examine the identification performance for the first rank candidate and retrieval performance for low penetration rates of the database, as we believe that the top candidates are the most important for any recognition process. We set to improve the retrieval performance on the FVC 2000 Db2a, FVC 2000 Db3a, FVC 2002 Db1a, and the modified NIST-4 database in this work.

## III. BASELINE RETRIEVAL SYSTEM

The proposed system is built on top of a baseline system. The baseline system is implemented according to the baseline descriptions in our recent work focusing on artificial sample generation [12], unless otherwise specified in Section III-A–III-C below. The baseline system is built up from a number of existing algorithms and techniques. It is composed of the following major modules:

- 1) orientation field estimation;
- 2) foreground/background segmentation;
- 3) registration point detection;
- 4) image enhancement;
- 5) feature extraction;
- 6) retrieval.

### A. Registration Point Detection

Registration point detection is a crucial stage in fingerprint retrieval. Different algorithms for singular or registration point detection [2], [31]–[33] have been used in the publications concerning fingerprint retrieval, but we do not know how well each of the algorithms can achieve. For the ease of comparison in the future, we simply exploit the widely used and well-tested PCASYS software to locate a reference point on each fingerprint image. Note that the software was originally designed for working with five-class classification on the NIST-4 database. However, the embedded R92 algorithm for registration point detection that renders satisfactory performance on the NIST-4 database can be singled out for the application to the FVC 2000 and FVC 2002 databases. The work by Senior [34] reported the performance of the PCASYS software.

### B. Feature Extraction

The modified FingerCode is employed as features in our experiments [2]. We filter an enhanced fingerprint image by Gabor filters in  $M(= 12)$  directions to capture finer details of the ridge structures. The filtering orientations are  $(0, \pi/12, 2\pi/12, \dots, 11\pi/12)$ . Unlike the original FingerCode, we define each feature cell as a square sector of size  $v \times v$  instead of the shape of radial sector. For the FVC 2000 and FVC 2002 databases, the *center of feature extraction* is exactly at the position of the registration point found by the PCASYS software, as opposed to shifting downwards by 40 pixels for the NIST-4 database [2]. This is due to the fact that many fingerprint images in the FVC databases have their registration point located close to the lower border of the frame.

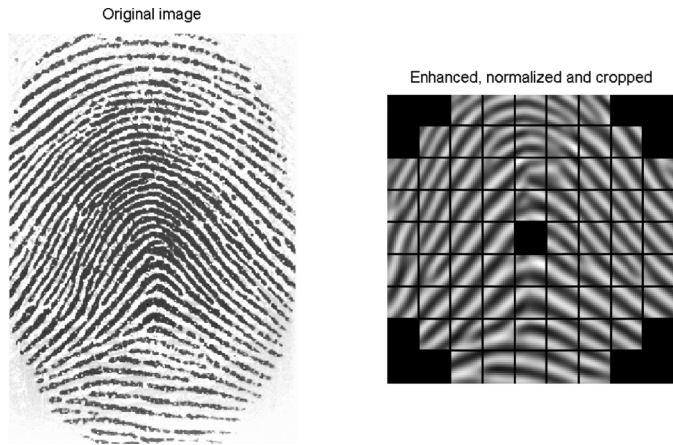


Fig. 1. Blocks of interest for feature extraction (68 blocks per Gabor-filtered image) on an image from the FVC 2000 Db2a.

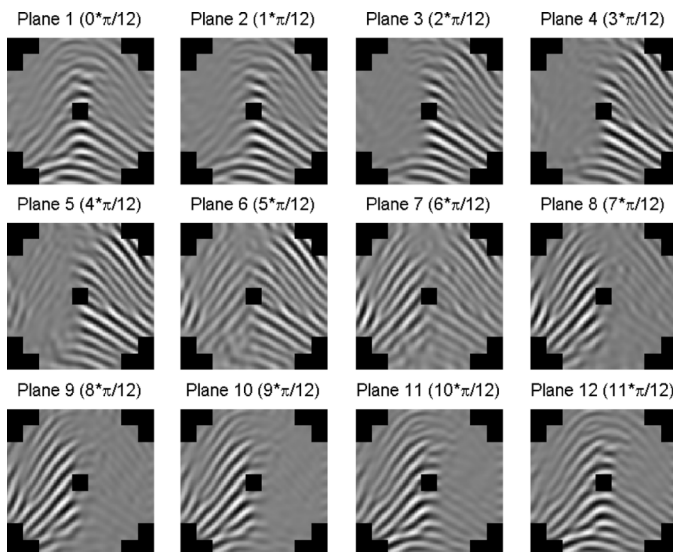


Fig. 2. Example set of Gabor-filtered fingerprint images (using the same fingerprint as in Fig. 1) in 12 directions.

We do not use any weighted feature as described in our previous work [12] to attenuate features situating far away from the registration point [21], mainly because the FVC databases do not provide enough fingers for accurate estimation of the weights. Variable transformation that takes the power  $P_v$  of each feature value is also not used [12]. A graphical illustration for defining the blocks of interest for feature extraction is shown in Fig. 1 and an example set of Gabor-filtered images in 12 directions is given in Fig. 2.

### C. Fingerprint Retrieval

We adhere to the Euclidean 1-NN classification approach in the baseline system implementation. Given a feature vector  $\mathbf{x} = (x_1, x_2, \dots, x_d)^T$  for a test sample, a feature vector  $\mathbf{y} = (y_1, y_2, \dots, y_d)^T$  for a sample in the database, and the two binary vectors  $\mathbf{z}_x = (z_{x1}, z_{x2}, \dots, z_{xd})^T$  and  $\mathbf{z}_y = (z_{y1}, z_{y2}, \dots, z_{yd})^T$  defining the validity of the features (with value of 0 for background or missing feature, and 1 for

foreground feature), the distance  $D_{xy}$  of the two fingerprints is computed by

$$D_{xy} = \sqrt{\frac{1}{\sum_{m=1}^d z_{xm}z_{ym}} \sum_{k=1}^d z_{xk}z_{yk}(y_k - x_k)^2}. \quad (1)$$

A commonly used retrieval method in pattern classification is the fixed order retrieval. Given a target penetration rate and a test image from an unknown finger, a list of probable fingers with the list length proportional to the penetration rate is produced based on the distance of the test sample to different fingers. It is noted that some other fingerprint retrieval methods with better nominal performance were proposed [21]. In the current work, though, we shall stick to the traditional fixed order retrieval method which is more technically sound and suitable for identification and retrieval task.

## IV. PROPOSED BAYES CLASSIFICATION FOR FINGERPRINT RETRIEVAL

As mentioned in Section I, fingerprint retrieval systems mostly adopt the 1-NN classifier due to the fact that there are only two and eight samples per finger in the NIST-4 and FVC databases, respectively, which are far from sufficient for training up more sophisticated classifiers. Despite its simplicity, the 1-NN classifier is inferior to many classifiers such as the Bayes classifier. Cover [35] has proven that the risk of 1-NN classification is bounded by twice the risk of the Bayes classifier, and it approaches the performance of the Bayes classifier asymptotically only when there is an infinite number of training samples.

In this paper, we propose to adopt the Bayes classifier assuming Gaussian statistics, and this requires the following issues to be addressed: 1) shortage of training samples; 2) curse of dimensionality leading to inaccurate statistical estimation and over-fitting; and 3) remedy of missing feature values. Our recently proposed fingerprint spatial modeling techniques allow us to generate several tenths of additional training samples from one genuine sample, directly solving the first issue partly.

Regarding the second issue, it is realized that the original feature dimension is very high, ranging from a few hundred to a thousand [12], [16], [17], [19]–[22], [24]. We implement a variant of the Fisher's linear discriminant for dimension reduction and a variant of the quadratic discriminant function (Bayes decision function assuming Gaussian statistics) for lowering estimation errors. Instead of simply adopting the standard formulations, we regularize some of the estimated matrices. In the Fisher's linear discriminant, we regularize the within-class scatter matrix  $S_W$  by adding a constant  $h_w^2$  to the diagonal elements of the matrix, while keeping the between-class scatter matrix  $S_B$  intact. In the quadratic discriminant function, we regularize the covariance matrix  $\sum_i$  in each class  $i$  (the term "class" here means the "identity of a finger," but *not* Henry class) by using a class-independent constant  $h^2$ , for which the technique is termed modified quadratic discriminant function 1 (MQDF1) in the literature [36], [37].

Regularization in both Fisher's linear discriminant and quadratic discriminant function may be required with reasonable constants, or else the identification and retrieval rates could drop. For optimal performance, these regularization constants have to be adjusted according to the number of training samples per finger and feature dimension.

The final issue to be resolved is the treatment of missing feature values. The segmentation algorithm defines the foreground and background areas of a fingerprint image, and the background features should not be used in training a classifier. There also exist quite a lot of fingerprint images in the FVC 2000 and FVC 2002 databases with their registration point locating very close to the edge of the image frame. In such cases, almost half of the feature values would be missing when we extract Gabor features around the registration point. The missing features can be ignored in distance-based  $k$ -NN classification—only those features available on both the training and test sample will be included in calculating the distance, as shown in (1).

In Bayes classification along with Fisher's linear discriminant, however, all features must be present in all training samples for the calculation of class statistics and feature projection, and in all test samples for feature projection and Bayes classification. We take the mean imputation approach to get around this pitfall in our current attempt [38]. Given a set of  $q$  training samples  $\{\mathbf{y}_i^1, \mathbf{y}_i^2, \dots, \mathbf{y}_i^q\}$  of class  $i$ , with each feature vector specified by  $\mathbf{y}_i^{q_0} = (y_1^{q_0}, y_2^{q_0}, \dots, y_d^{q_0})^T$  and validity of feature specified by  $\mathbf{z}_i^{q_0} = (z_1^{q_0}, z_2^{q_0}, \dots, z_d^{q_0})^T$ , we construct a relaxed class mean vector  $\mathbf{m}_i$

$$\begin{aligned} \mathbf{m}_i &= (m_1^i, m_2^i, \dots, m_d^i)^T \\ &= \left( \frac{1}{\sum_{t=1}^q z_1^t} \sum_{u=1}^q y_1^u z_1^u, \frac{1}{\sum_{t=1}^q z_2^t} \sum_{u=1}^q y_2^u z_2^u, \dots, \right. \\ &\quad \left. \frac{1}{\sum_{t=1}^q z_d^t} \sum_{u=1}^q y_d^u z_d^u \right)^T. \end{aligned} \quad (2)$$

We further define for the training samples a global relaxed mean vector  $\mathbf{m}$  over all  $m_{d_0}^i \neq \infty$  by

$$\begin{aligned} \mathbf{m} &= (m_1, m_2, \dots, m_d)^T \\ &= (\overline{m_1^i}, \overline{m_2^i}, \dots, \overline{m_d^i})^T. \end{aligned} \quad (3)$$

In the case where  $\sum_{t=1}^q z_{d_0}^t = 0$ , we simply set the mean value of the feature ( $m_{d_0}^i$ ) to be  $m_{d_0}$  which is the global mean feature value over all classes in the database. With the relaxed mean vector  $\mathbf{m}_i$ , for all  $\mathbf{y}_i^{q_0}$  with zero(s) in  $(z_1^{q_0}, z_2^{q_0}, \dots, z_d^{q_0})^T$ , the value of  $y_{d_0}^{q_0}$  for any feature with  $z_{d_0}^{q_0} = 0$  is substituted by the corresponding value of  $m_{d_0}^i$ . By this substitution, we can apply on the training samples the Fisher's linear discriminant, feature projection and train the Bayes classifier as usual.

Given a test sample  $\mathbf{x} = (x_1, x_2, \dots, x_d)^T$  with validity specified by  $\mathbf{z} = (z_1, z_2, \dots, z_d)^T$ , if any of the  $z_{d_0}$ 's is zero, then the corresponding  $x_{d_0}$  is assumed to be  $m_{d_0}$ . This substitution is inevitable since we do not know what finger the unknown sample belongs to prior to classification.

It should be pointed out that substituting a mean class feature value to the missing feature on training samples would flatten

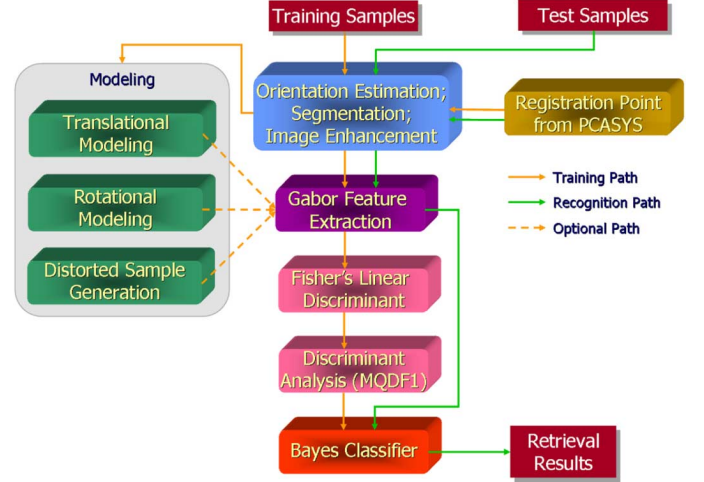


Fig. 3. Recognition flowchart of the newly proposed fingerprint retrieval system featuring Fisher's linear discriminant and the Bayes classifier.

the shape of the cluster for that class in the feature space. In particular, the variance along the dimension of missing feature, hence the related eigenvalues, would be smaller than what is expected when the "real" feature value is available. Other sophisticated imputation methods may be adopted to diminish such errors.

Our proposed retrieval system is made possible after solving the above issues. The new recognition flowchart with Fisher's linear discriminant and Bayes classification is drawn in Fig. 3.

## V. PERFORMANCE EVALUATION

### A. System Setup

We evaluate the performance of our proposed system on the FVC 2000 Db2a, FVC 2000 Db3a, FVC 2002 Db1a, and the NIST-4 database with reduced cardinality. A set of the FVC 2000 or FVC 2002 database contains 800 images taken from 100 fingers (eight instances per finger), while a set of the modified NIST-4 database contains 2366 images taken from 1183 fingers (two instances per finger) extracted from a predefined manner [12] that follows the natural proportion of Henry class distribution.

The small number of samples per finger limits the size of the training set. It is a normal practice in machine learning that the number of training samples should be more than that of, or at least comparable to, the test samples, especially for Bayes classification where a large amount of samples should be used for training. We, therefore, adjust the partitioning scheme on the FVC 2000 and FVC 2002 databases in our experiments.

The retrieval rate, defined as the percentage of test fingerprints whose actual identities are retrieved successfully, is reported together with a penetration rate, the average portion of fingers in the database that has been retrieved. The retrieval error rate is defined as  $(100\% - \text{retrieval rate})$ . The identification rate is the percentage of test fingerprints which are correctly classified as their nominal finger from the classifier's first rank output. We are interested in the identification/retrieval performance at low penetration rates by employing the traditional fixed order retrieval method.

TABLE I  
GENERAL PARAMETERS ADOPTED IN EXPERIMENT 1–4

Parameter	FVC 2000 Db2a	FVC 2000 Db3a	FVC 2002 Db1a	Modified NIST-4
$W$	256	448	388	512
$H$	364	478	374	480
$c$	100			1,183
$d$	$68 \times 12 = 816$			
$d_r$	99			99, 200 or 816
$M$	12			
$v$	15	25	17	39
$R$	4			
$P_v$	1.0			
$a_x$	Random number $\pm [0.01, 0.40]$			
$a_y$	Random number $\pm [0.01, 0.40]$			
$T_s$	0.05			

TABLE II  
MODELING PARAMETERS FOR THE ARTIFICIAL TRAINING SAMPLES  
GENERATED PER GENUINE SAMPLE IN EXPERIMENT 1–4

Sample number	Translational change (number of pixels)	Rotational change (angle)	Plastic distortion model
1 – 4	$[(\pm 5, 0), (0, \pm 5)]$	No	Not applied
5 – 8	$[(\pm 15, 0), (0, \pm 15)]$	No	Not applied
9 – 12	$[(\pm 10, +10), (\pm 10, -10)]$	No	Not applied
13 – 14	No	$[\pm \pi/12]$	Not applied
15 – 16	No	$[\pm \pi/6]$	Not applied
17 – 24	No	No	Applied

We define  $c$  as the number of fingers,  $d$  as the original feature dimension, and  $d_r$  as the reduced dimension. The parameters discussed in Sections III and IV and in [12] which will be used in all of the following experiments are given in Table I. Other parameters that are not mentioned in this paper have been kept unchanged and can be found in [12]. When artificial samples are produced, the modeling types and parameters are defined in Table II and the total number of generated samples is always 24. It should be mentioned that for the FVC databases, the number of fingers  $c$  to be used is always 100, so the maximum number of projection planes in Fisher's linear discriminant is 99. For the NIST-4 database, as  $c$  is very large, both the within-class scatter matrix  $S_W$  and the between-class scatter matrix  $S_B$  should practically be of full rank.

The experiments are conducted on a PC equipped with an Intel Core2Duo E8400 CPU and 4-GB random access memory. The simulator is written in Matlab language and is executed under the provided run-time environment. In Experiment 1, we generate additional samples to train both the 1-NN classifier and the Bayes classifier, and observe the performance gain on top of the baseline system without extra samples. The effect of the number of genuine training samples and test samples on the accuracy of the Bayes classifier is examined in Experiment 2.

In Experiment 3, we compare the retrieval performance of our baseline and proposed systems with other reported figures in the literature for all databases under test. In Experiment 4, the degree of dimension reduction to the retrieval performance is evaluated on the modified NIST-4 database.

In Experiments 1 and 2, the retrieval error rates are reported by averaging two sets of simulation results to smooth out the instability caused by relatively small number of classes in the databases. For instance, when four samples are used for training, the first simulation exploits fingerprint instances 1–4 as training samples and instances 5–8 as test samples. The second simulation rotates the two sample sets for cross-validation. Refer to Table III for other combinations of samples.

## B. Experimental Results

*Experiment 1:* The first experiment concerns with the improvement on retrieval rates by applying artificial sample generation, comparing between 1-NN and Bayes classification. Experiments are conducted on the FVC 2000 Db2a, FVC 2000 Db3a, and FVC 2002 Db1a with four genuine training samples and four test samples per finger. In the experiment,  $h_w^2$  is fixed at the 600th eigenvalue of the matrix  $S_W$  and  $h^2$  is iteratively searched for a near-optimal value. We compare the identification

TABLE III  
INSTANCE NUMBER OF TRAINING AND TEST SAMPLES BEING USED IN DIFFERENT SETS OF SIMULATION

	<b>1 training sample; 7 test samples</b>	<b>4 training samples; 4 test samples</b>	<b>6 training samples; 2 test samples</b>
Training sample instance number (first simulation)	1	1 – 4	1 – 6
Test sample instance number (first simulation)	2 – 8	5 – 8	7 – 8
Training sample instance number (second simulation)	8	5 – 8	3 – 8
Test sample instance number (second simulation)	1 – 7	1 – 4	1 – 2
Regularization constant $h_w^2$	1st eigenvalue of $S_W$	600th eigenvalue of $S_W$	0

TABLE IV  
COMPARISON ON AVERAGE RETRIEVAL ERROR RATES AT VARIOUS PENETRATION RATES OF THE FVC 2000 Db2a, FVC 2000 Db3a, AND FVC 2002 Db1a BY 1-NN AND BAYES CLASSIFICATIONS

<b>Database</b>	<b>Rank (Penetration rate)</b>	<b>1-NN classification without extra samples</b>	<b>1-NN classification with extra samples (Error reduction)</b>	<b>Bayes classification with extra samples (Error reduction)</b>
FVC 2000 Db2a	1 (1%)	8.0%	6.0% (-25.0%)	<b>4.9% (-39.1%)</b>
	1 – 2 (2%)	6.0%	4.6% (-22.9%)	<b>4.1% (-31.3%)</b>
	1 – 5 (5%)	3.4%	<b>2.5% (-25.9%)</b>	3.1% (-7.4%)
FVC 2000 Db3a	1 (1%)	22.1%	17.2% (-22.0%)	<b>15.1% (-31.6%)</b>
	1 – 2 (2%)	16.1%	13.5% (-16.3%)	<b>11.9% (-26.4%)</b>
	1 – 5 (5%)	10.9%	<b>9.0% (-17.2%)</b>	9.9% (-9.2%)
FVC 2002 Db1a	1 (1%)	17.4%	10.2% (-41.0%)	<b>7.1% (-59.0%)</b>
	1 – 2 (2%)	14.5%	7.4% (-49.1%)	<b>5.5% (-62.1%)</b>
	1 – 5 (5%)	10.4%	5.0% (-51.8%)	<b>4.5% (-56.6%)</b>

and retrieval performance on the databases at first rank output, first and second rank output, and 5% penetration rate from the 1-NN classifier and Bayes classifier. Table IV reports the simulation results obtained.

With four genuine training and test samples, the identification rate of the 1-NN classifier without extra samples are about 92%, 78%, and 83% on the FVC 2000 Db2a, FVC 2000 Db3a, and FVC 2002 Db1a, respectively. The error rates for first rank output have been reduced by about 22% to 41% by generating additional samples to train the 1-NN classifier, while they have been reduced by about 32% to 59% by our proposed Bayes classification scheme. The quality of fingerprint images in the three databases is different, leading to very different retrieval error rates. Having said that, a similar trend of error reduction after applying artificial sample generation and Bayes classification can be clearly perceived for all three databases, edging the corresponding 1-NN classification.

*Experiment 2:* This experiment focuses on the retrieval performance of the Bayes classifier by using variable amount of training samples. The experiment is carried out on the FVC 2000 Db2a with the following combinations of training and test samples: 1) one training/seven test samples; 2) four training/four test samples; and 3) six training/two test samples.

It has been mentioned in Section IV that different regularization constants  $h_w^2$  and  $h^2$  should be specified for different numbers of training samples. Generally speaking, the more training

samples available, the smaller the constants can be as we are more confident about the overall class statistics. The constants  $h_w^2$  for combinations 1), 2), and 3) are given in Table III, while  $h^2$  is iteratively searched for a near-optimal value. We do not consider the necessity to adjust the regularization constants a problem in real-life situation, because we would have already known the actual number of training samples per finger in most applications, such that the regularization constants can be reasonably determined prior to the training stage of the system. Table V presents the averaged results for the three groups of sample set by Bayes classification, with reference to the results obtained by 1-NN classification with the same amount of additional samples.

From Table V, the greater number of (genuine) training samples available, the lower the retrieval error rates at all penetration rates. Moreover, Bayes classification outperforms 1-NN classification when more genuine training samples are available, but not in the case of only one genuine sample. This is an expected result because we do not formally deal with the missing features, thereby slightly sacrificing the power of the Bayes classifier. The problem of missing features diminishes when more training samples are available because the substitution of global mean feature value would be less frequent than that with only one genuine training sample. Although Bayes classification with one genuine training sample together with the artificially generated samples is inferior to 1-NN classification with the same



TABLE V  
COMPARISON ON AVERAGE RETRIEVAL ERROR RATES AT VARIOUS PENETRATION RATES OF THE FVC 2000 Db2a BY 1-NN AND BAYES CLASSIFICATIONS WITH DIFFERENT COMBINATIONS OF TRAINING AND TEST SAMPLES

Classifier	Rank (Penetration rate)	1 training sample; 7 test samples	4 training samples; 4 test samples	6 training samples; 2 test samples
Number of artificial samples per genuine sample		24		
Bayes	1 (1%)	15.0%	4.9%	2.7%
	1 – 2 (2%)	12.3%	4.1%	2.0%
	1 – 5 (5%)	8.8%	3.1%	1.7%
1-NN	1 (1%)	12.6%	6.0%	3.0%
	1 – 2 (2%)	9.9%	4.6%	2.2%
	1 – 5 (5%)	7.0%	2.5%	2.0%

number of training samples, the former one may still be a good trade-off between speed and accuracy, as will be revealed in Section V-C on timing analysis.

*Experiment 3:* It is informative to compare the retrieval performance of our system on the FVC 2000 Db2a, FVC 2000 Db3a, FVC 2002 Db1a, and the modified NIST-4 databases with those reported in the literature. In this experiment, the first instance of the fingerprints forms the training set while the rest form the test set (without rotation of sample set) for the case of one genuine training sample to facilitate comparisons. Fig. 4 depicts the retrieval performance of the baseline method (the Euclidean 1-NN fixed-order retrieval without generating additional samples) and the proposed Bayes classification method, compared with some reported retrieval error rates. Note that the results may not be directly comparable with each other. In particular, the work by Bhanu and Tan [18], Liang *et al.* [26], [27], Feng and Cai [28], and Shuai *et al.* [29] were based on minutiae matching or its variants, and some results were obtained by the extended FVC and NIST-4 databases.

From Fig. 4(a) and (c), it is shown that Bayes classification performs better under low penetration rates in the FVC 2000 Db2a and under all penetration rates in the FVC 2002 Db1a compared with the baseline system. From Fig. 4(b), Bayes classification is still slightly better than the baseline in the FVC 2000 Db3a, but not as good as the results obtained by Jiang *et al.* [21]. From Fig. 4(d) for the NIST-4 database, our Bayes approach performs the second best among all published results between 0.0845% (first rank) to 20% penetration rate [16]–[19], [22], [23], verifying that the technique is scalable to large databases.

*Experiment 4:* The final experiment concerns with the degree of dimension reduction to the actual retrieval error rates attained by the Bayes classifier. The experiment is done on the modified NIST-4 database only, with the first instance of the fingerprints forming the training set. We report the retrieval error rates for 99-D classification, 200-D classification and 816-D classification (without Fisher’s linear discriminant), as drawn in Fig. 5. No significant change of retrieval performance can be noticed over a wide range of reduced dimensional space.

### C. Timing Analysis

Besides offering better retrieval performance in most of the simulation environments, another major advantage of

employing Bayes classification is to reduce the lengthy classification time incurred by the 1-NN classifier, especially when the total number of training samples is large. Let the number of genuine training samples in each class  $i$  be  $n_i$ . Since  $n_1 = n_2 = \dots = n_c (= n)$  for all of the experiments in this paper, assuming that  $g$  samples are generated per genuine training sample, the total number of training samples is  $(g + 1)cn$ . In terms of timing complexity of classification of a single test sample, the 1-NN classifier is of the order  $\mathcal{O}(gcmd)$  while the Bayes classifier, following the formulation in (4), is of the order  $\mathcal{O}(cd_r^2)$

$$p'_i(\mathbf{x}_0) = \mathbf{x}_0^T A_i \mathbf{x}_0 + \mathbf{b}_i^T \mathbf{x}_0 + c_i \quad (4)$$

where  $\mathbf{x}_0$  is the projected feature vector with dimension  $d_r$  and  $A_i, \mathbf{b}_i, c_i$  are constants in each class  $i$ .

Bayes classification involves application of (4)  $c$  times for classifying one test sample which, unlike 1-NN classification that requires invoking (1)  $gcn$  times. With the Bayes classifier, the number of training samples per finger does not affect the timing complexity in classification as the class statistics are represented by a cluster with mean and variance in the feature space. Furthermore, the dimension reduction by Fisher’s linear discriminant implies that the vector computation time can be reduced since the new feature dimension  $d_r$  in our experiments on the FVC databases is just about one eighth of the original feature dimension  $d$ .

Table VI presents the timing complexity of our simulation platform to classify one unknown sample in the FVC database, averaged over 1200 samples to smooth out the variations. Results are compared among 1-NN classifier without extra samples, 1-NN classifier with extra samples (24 artificial samples per genuine sample) and Bayes classifier with extra samples. The number of genuine training samples is four per finger, with a total of 100 fingers. We also estimate the total number of mathematical operations (additions and multiplications) required by the two classifiers following the implementation of (1) and (4). For the Bayes classifier, the number of operations for projecting the original feature vector of the test sample to the reduced feature space should also be included.



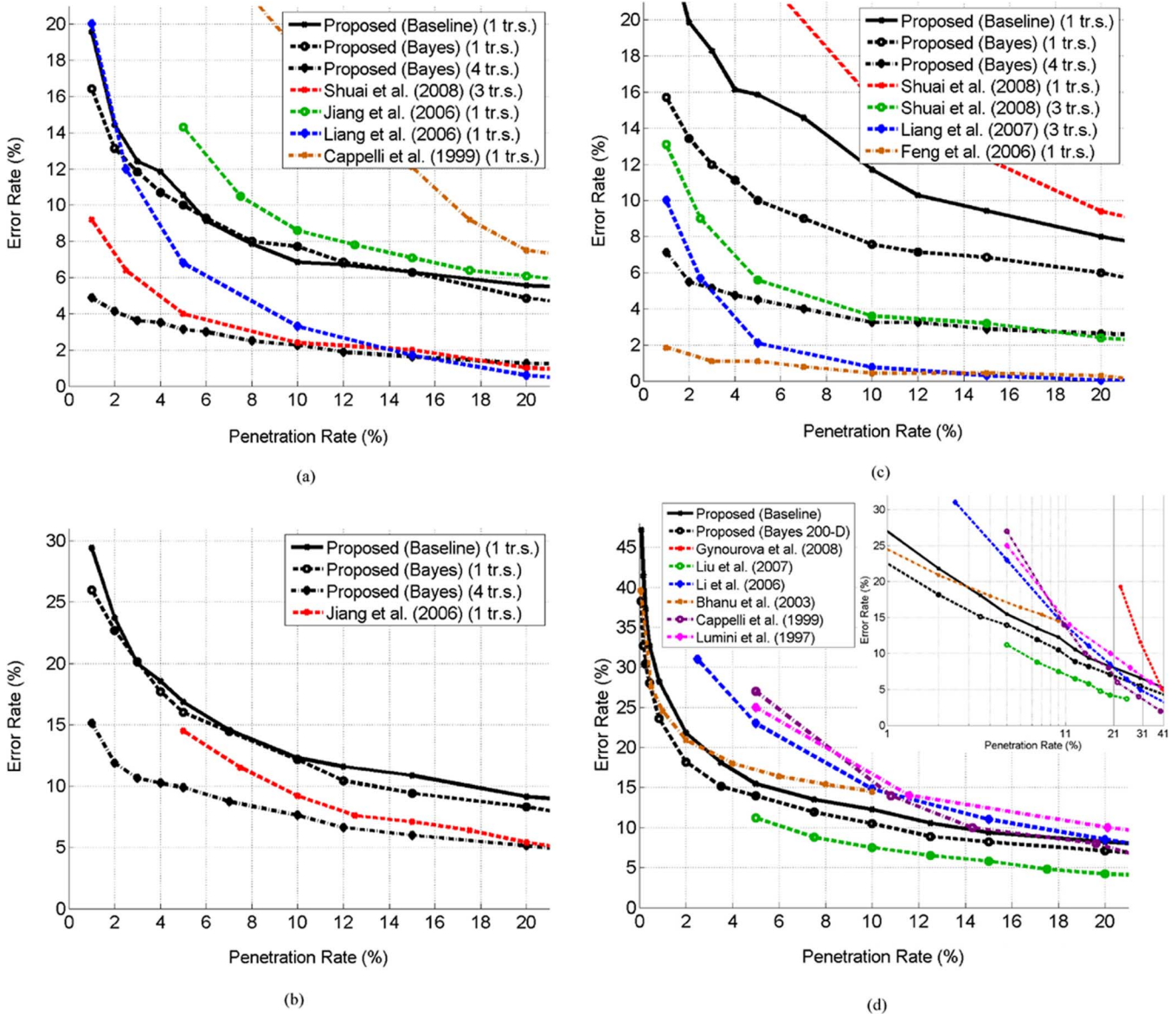


Fig. 4. Retrieval performance curves for 1-NN and Bayes classifications, compared with other published results. The abbreviation “tr.s.” denotes the term “training sample(s)”. (a) Results for the FVC 2000 Db2a. (b) Results for the FVC 2000 Db3a. (c) Results for the FVC 2002 Db1a. (d) Results for the modified NIST-4 database.

In this example, the Bayes classifier, with the advantage of additional samples, runs twice faster than the 1-NN classifier without any additional samples. From the results in Tables IV and VI, we are in fact taking the best of both worlds, accuracy and time, at no extra cost. The 1-NN classifier with extra samples, as expected, requires the largest amount of time to classify an unknown sample, directly proportional to the number of extra samples being generated. The estimation of mathematical operations and CPU time are only for brief reference, as the actual programmed instructions together with the Matlab’s data structures and functional abstractions can greatly affect the number of operations executed and the simulation time.

Nevertheless, Table VI only shows the speed improvement from 1-NN to Bayes classification under our simulation framework. The actual “speed of retrieval” is also dependent on other factors, such as feature dimension, method of feature extraction,

and whether database clustering is applied. Hence, our approach may not necessarily be working faster than other reported techniques. Although a direct comparison of retrieval speed is not possible, Table VII compares our work with two other proposals that aim at high retrieval efficiencies.

It can be realized that retrieval efficiency can also be enhanced by reducing the feature dimension drastically (as in Cappelli *et al.* [24]) or by implementing database clustering (as in Jiang *et al.* [22]). With database clustering, a query fingerprint is only compared with cluster prototypes prior to the computation of the similarity to the fingers inside the selected clusters. Eventually, the lower the penetration rate, the smaller is the required search space which would be much smaller than the entire database. Our Bayes classification approach is thus only focusing on one of the aspects (feature dimension) where retrieval speed can potentially be improved.

TABLE VI  
COMPLEXITY AND TIMING ANALYSIS FOR CLASSIFYING ONE TEST SAMPLE IN THE FVC DATABASE BY 1-NN AND BAYES CLASSIFICATIONS  
(TAKE  $g = 24$ ,  $c = 100$ ,  $n = 4$ ,  $d = 816$  AND  $d_r = 99$ )

	1-NN classification without extra samples	1-NN classification with extra samples	Bayes classification with extra samples
Complexity	$O(cnd)$	$O(gcnd)$	$O(cd_r^2)$
Number of additions (projection)	0	0	80,685
Number of additions (classification)	978,400	24,460,000	990,000
Number of multiplications (projection)	0	0	80,784
Number of multiplications (classification)	1,306,400	32,660,000	999,900
Number of mathematical operations	2,284,800	57,120,000	2,151,369
CPU time required for projection and classification instructions	18.3 ms	454.3 ms	7.2 ms

#### D. Discussions and Elaborations on Experimental Results

From the results in Sections V-B and V-C, it can be concluded that, when the Bayes classifier is only trained with one genuine sample and the associated artificial samples, then it delivers better retrieval performance (especially at low penetration rates) than the 1-NN classifier without artificial samples, but with comparable execution time. If efficiency is not a major consideration, then one can simply train the 1-NN classifier with the artificial samples to achieve better performance than the Bayes classifier. However, when the Bayes classifier is trained with more genuine samples together with the artificial samples, then it delivers higher accuracy in shorter execution time than the 1-NN counterpart. Therefore, Bayes classification should be the choice over 1-NN when more training samples are available in a given database.

In our implementation, the 1-NN classifier works on feature vectors with an original dimension of 816, while the Bayes classifier works with a reduced dimension of 99 in most of the cases. It can be argued that if the 1-NN classifier also worked on a 99-D feature space, its speed would be much higher than the Bayes classifier. However, the problem is that if we directly extract a 99-D feature vector from the raw image, its discrimination power would be less than that of an 816-D feature vector. From our observation, the retrieval rates drop by about 10% for the first few top rank candidates when we train the 1-NN classifier with a feature dimension of approximately 100.

In practical application, it is possible that new fingers are added into the database incrementally. This is not a problem for the 1-NN classifier, but with the Bayes approach, it is unlikely that the training can be performed each time a new record is inserted. In this case, the first option is to discard the dimension reduction stage in operating the Bayes classifier. Then inserting a new record is nothing more than calculating the mean vector and covariance matrix for that record and appending them to the classifier. From the results of Experiment 4 (Fig. 5), such approach would not degrade the classification performance with regularizations in place, but the computational efficiency would

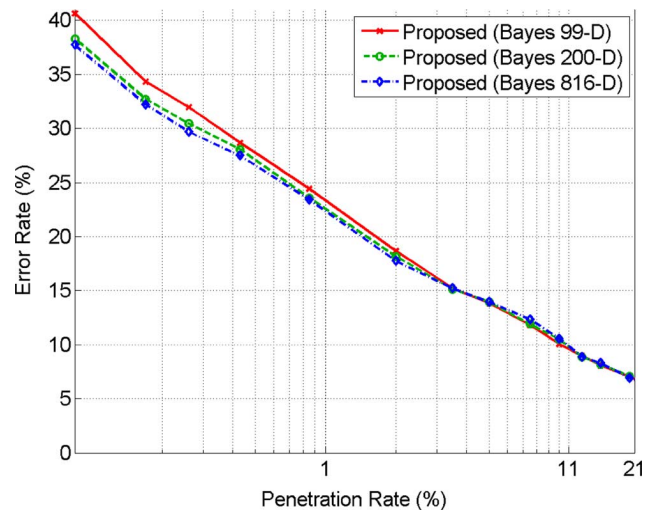


Fig. 5. Retrieval performance on the modified NIST-4 database by Bayes classification with different reduced dimensions. The  $x$ -axis is drawn in log scale for better visual quality.

be compromised. The second option is to fix the projection vectors from Fisher's linear discriminant after a complete training process, assuming that adding an extra record would not severely affect the output of linear discriminant. The Bayes classifier can then be trained normally without retouching the old records. When too many new records are inserted since the last complete training, Fisher's linear discriminant and subsequent classifier training should be carried out again by using the stored mean vector and covariance matrix of each class.

#### VI. CONCLUDING REMARKS

The retrieval performance of the baseline system is satisfactory compared with other published results on the FVC and NIST-4 databases. When compared with orientation field features and complex-filtered features, it is evident from Experiment 3 that Gabor features give better results as far as fingerprint retrieval is concerned.

TABLE VII  
COMPARISON ON THE COMPUTATIONAL COMPLEXITY OF OUR 1-NN AND BAYES CLASSIFICATION METHODS  
TOGETHER WITH OTHER EXISTING APPROACHES

	<b>1-NN classification without extra samples</b>	<b>1-NN classification with extra samples</b>	<b>Bayes classification with extra samples</b>	<b>Jiang et al. [22] (NIST-4)</b>	<b>Cappelli et al. [24] (FVC 2000)</b>
Method to enhance the efficiency	N/A	N/A	Fisher's linear discriminant	Database clustering	Multi-space K-L transform
Feature extracted	Gabor feature	Gabor feature	Gabor feature	Orientation feature and average ridge distance	Orientation feature
Original feature dimension	816-D	816-D	816-D	156-D + 1-D	1,680-D
Final feature dimension	816-D	816-D	99-D for FVC; 99-D, 200-D or 816-D for NIST-4	156-D + 1-D	9-D
Search space at 5% penetration rate	$c$	$(g+1)c$	$c$	Approximately $0.15c$	$c$
Search space at 20% penetration rate	$c$	$(g+1)c$	$c$	Approximately $0.33c$	$c$

Our previously proposed spatial modeling techniques have made the retrieval process more reliable. We tackle the problem of speed by proposing the use of Fisher's linear discriminant and Bayes classification with the methods for regularization and handling of missing features. Moreover, the timing analysis in Section V-C shows that Bayes classification with any number of training samples per finger runs faster than 1-NN classification with four training samples per finger. The execution time of 1-NN classification increases linearly with the number of training samples per finger, but remains the same for Bayes classification. As the speed of Bayes classification does not depend on the number of training samples per finger, one can capture many instances of the same finger for building up a large but comprehensive database without the fear of lowering the retrieval speed and accuracy.

The experimental results also show that regularizing the covariance matrices in the Fisher's linear discriminant and Bayes classifier is a useful strategy. Taking half of the samples for training and the remaining half for testing, Bayes classification, compared with the baseline system, has helped reduce the first rank error rates by 39.1%, 31.6%, and 59.0% in the FVC 2000 Db2a, FVC 2000 Db3a, and FVC 2002 Db1a, respectively. The retrieval rates for all four databases under test are very satisfactory compared with other published figures, such that the proposed techniques could be of practical uses in security, civilian, and forensic applications.

Nevertheless, the weakness of the proposed Bayes approach with Gaussian statistics is that it does not perform particularly well with a single genuine sample per finger. It is probably an excellent choice only when more training samples are available.

Moreover, when a large number of features is missing, the performance of the quadratic classifier also degrades, and further research is needed for improvement.

#### ACKNOWLEDGMENT

The authors would like to express their sincere thanks to the reviewers for their helpful comments.

#### REFERENCES

- [1] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*, 2nd ed. New York: Springer, 2009.
- [2] A. K. Jain and L. Hong, "A multichannel approach to fingerprint classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 4, pp. 348–359, Apr. 1999.
- [3] K. Karu and A. K. Jain, "Fingerprint classification," *Pattern Recognit.*, vol. 29, no. 3, pp. 389–404, 1996.
- [4] R. Cappelli, D. Maio, D. Maltoni, and L. Nanni, "A two-stage fingerprint classification system," in *Proc. 2003 ACM SIGMM Multimedia Biometrics Methods and Applications Workshop*, Berkeley, CA, 2003, pp. 95–99.
- [5] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based fingerprint matching," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 846–859, May 2000.
- [6] A. K. Jain, L. Hong, and R. Bolle, "On-line fingerprint verification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 4, pp. 302–314, Apr. 1997.
- [7] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. Hoboken, NJ: Wiley, 2001.
- [8] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, "FVC2000: Fingerprint verification competition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 402–412, Mar. 2002.
- [9] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, "FVC2002: Second fingerprint verification competition," in *Proc. 16th Int. Conf. Biometric Authentication*, Quebec, Canada, 2002, pp. 811–814.

- [10] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, "FVC2004: Third fingerprint verification competition," in *Proc. 17th Int. Conf. Biometric Authentication*, Cambridge, U.K., 2004, pp. 1–7.
- [11] C. I. Watson and C. L. Wilson, NIST Special Database 4: Fingerprint Database National Institute of Standards and Technology, Tech. Rep., 1992.
- [12] K. C. Leung and C. H. Leung, Fingerprint retrieval by spatial modeling and distorted sample generation, unpublished.
- [13] A. Almansa and L. Cohen, "Fingerprint image matching by minimization of a thin-plate energy using a two-step algorithm with auxiliary variables," in *IEEE Workshop Applications of Computer Vision*, 2000, pp. 35–40.
- [14] R. Cappelli, D. Maio, and D. Maltoni, "Modelling plastic distortion in fingerprint images," in *Proc. Second Int. Conf. Advances in Pattern Recognition*, Rio de Janeiro, Brazil, 2001, pp. 369–376.
- [15] A. Senior and R. Bolle, "Improved fingerprint matching by distortion removal," *IEICE Trans. Inf. Systems E84-D*, no. 7, pp. 825–832, 2001.
- [16] A. Lumini, D. Maio, and D. Maltoni, "Continuous versus exclusive classification for fingerprint retrieval," *Pattern Recognit. Lett.*, vol. 18, no. 10, pp. 1027–1034, 1997.
- [17] R. Cappelli and A. Lumini, "Fingerprint classification by directional image partitioning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 402–421, May 1999.
- [18] B. Bhanu and X. Tan, "Fingerprint indexing based on novel features of minutiae triplets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 616–622, May 2003.
- [19] J. Li, W.-Y. Yau, and H. Wang, "Continuous fingerprints classification by symmetrical filters," in *Proc. 2006 ACM Symp. Information, Computer and Communications Security*, Taipei, Taiwan, 2006, pp. 357–357.
- [20] M. Liu, X. Jiang, and A. C. Kot, "Fingerprint retrieval by complex filter responses," in *Proc. 18th Int. Conf. Pattern Recognition*, Hong Kong, 2006, pp. 1042–1045.
- [21] X. Jiang, M. Liu, and A. C. Kot, "Fingerprint retrieval for identification," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 4, pp. 532–542, Dec. 2006.
- [22] M. Liu, X. Jiang, and A. C. Kot, "Efficient fingerprint search based on database clustering," *Pattern Recognit.*, vol. 40, no. 6, pp. 1793–1803, 2007.
- [23] A. Gyaourova and A. Ross, "A novel coding scheme for indexing fingerprint patterns," *Lecture Notes Comput. Sci.*, vol. 5342, pp. 755–764, 2008.
- [24] R. Cappelli, D. Maio, and D. Maltoni, "Indexing fingerprint databases for efficient 1 : N matching," in *Proc. Sixth Int. Conf. Control, Automation, Robotics and Vision*, Singapore, 2000.
- [25] J. De Boer, A. M. Bazen, and S. H. Gerez, "Indexing fingerprint databases based on multiple features," in *Proc. 12th Annual Workshop on Circuits, Systems and Signal Processing (ProRISC)*, Veldhoven, The Netherlands, 2001.
- [26] X. Liang, T. Asano, and A. Bishnu, "Distorted fingerprint indexing using minutia detail and Delaunay triangle," in *Proc. Third Int. Symp. Voronoi Diagrams in Science and Engineering*, Alberta, Canada, 2006, pp. 217–223.
- [27] X. Liang, A. Bishnu, and T. Asano, "A robust fingerprint indexing scheme using minutia neighborhood structure and low-order Delaunay triangles," *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 4, pp. 721–733, Dec. 2007.
- [28] J. Feng and A. Cai, "Fingerprint indexing using ridge invariants," in *Proc. 18th Int. Conf. Pattern Recognition*, Hong Kong, 2006, pp. 433–436.
- [29] X. Shuai, C. Zhang, and P. Hao, "Fingerprint indexing based on composite set of reduced SIFT features," in *Proc. 18th Int. Conf. Pattern Recognition*, Florida, 2008, pp. 1–4.
- [30] V. A. Sujan and M. P. Mulqueen, "Fingerprint identification using space invariant transforms," *Pattern Recognit. Lett.*, vol. 23, no. 5, pp. 609–619, 2002.
- [31] K. Nilsson and J. Bigun, "Localization of corresponding points in fingerprints by complex filtering," *Pattern Recognit. Lett.*, vol. 24, no. 13, pp. 2135–2144, 2003.
- [32] G. T. Candela, P. J. Grother, C. I. Watson, R. A. Wilkinson, and C. L. Wilson, PCASYS—A Pattern-Level Classification Automation System for Fingerprints National Institute of Standards and Technology, Tech. Rep. NISTIR 5647, 1995.
- [33] Q. Zhang and H. Yan, "Fingerprint classification based on extraction and analysis of singularities and pseudo ridges," *Pattern Recognit.*, vol. 37, no. 11, pp. 2233–2243, 2004.
- [34] A. Senior, "A combination fingerprint classifier," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 10, pp. 1165–1174, Oct. 2001.
- [35] T. M. Cover, "Estimation by the nearest neighbor rule," *IEEE Trans. Inf. Theory*, vol. IT-14, no. 1, pp. 50–55, Jan. 1968.
- [36] F. Kimura, K. Takashina, S. Tsuruoka, and Y. Miyake, "Modified quadratic discriminant functions and the application to Chinese character recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 1, pp. 149–153, Jan. 1987.
- [37] K. C. Leung and C. H. Leung, "Recognition of handwritten Chinese characters by combining regularization, Fisher's discriminant and distorted sample generation," in *Proc. 10th Int. Conf. Document Analysis and Recognition*, Barcelona, Spain, 2009, pp. 1026–1030.
- [38] A. Farhangfar, L. Kurgan, and J. Dy, "Impact of imputation of missing values on classification error for discrete data," *Pattern Recognit.*, vol. 41, no. 12, pp. 3692–3705, 2008.



**K. C. Leung** received the B.Eng. degree in computer engineering from the University of Hong Kong. He is currently working toward the Ph.D. degree in the Department of Electrical and Electronic Engineering at the same University.

His research interests include pattern recognition and computer vision.



**C. H. Leung** received the B.Sc. (Eng.) and Ph.D. degrees from the University of Hong Kong, and the M.Eng. degree from McGill University, all in electrical engineering.

He has been teaching at the University of Hong Kong since 1986. His interests include pattern recognition and electronic circuits.