

Summary Page

Title:

Spectral Eigenfeatures for Effective DP Matching in Fingerprint Recognition

Authors:

Boris Danev, Toshio Kamei

Corresponding author:

Toshio Kamei

Address:

Media and Information Research Laboratories, NEC Corporation
1753, Shimonumabe, Nakahara, Kawasaki, Kanagawa 211-8666, Japan
Tel: +81 44 431 7713
Fax: +81 44 431 7589
Email: t-kamei@cb.jp.nec.com

Keywords:

Fingerprints, Eigenfeatures, Dynamic Programming

Answers to questions:**1. What is the original contribution of this work?**

The novelty of our work consists in introducing a two-step PCA for extracting spectral eigenfeatures in fingerprints that can be effectively matched by Dynamic Programming (DP).

2. Why should this contribution be considered important?

Our method can be embedded in small consumer devices such as laptops, PDA, USB keys due to its high accuracy and very compact representation of the fingerprint. It is suitable for small size line sensors as it requires very small region for feature extraction.

3. What is the most closely related work by others and how does this work differ?

Matsumoto et al. proposed a fingerprint verification method based on Dynamic Programming matching of spectral features. In this work, spectral features such as Linear Prediction Coefficients (LPC) were extracted from each image line and matched with Dynamic Programming.

Our work differs in proposing statistical analysis (PCA) for feature extraction coupled with an effective way of applying DP matching for eigenfeatures. Applying DP matching for eigenfeatures is usually difficult due to the loss of topographical information during feature extraction.

4. How can other researchers make use of the results of this work?

Our work provides a framework for applying more advanced statistical analysis techniques such as Linear Discriminant Analysis and Kernel methods to further improve the recognition accuracy.

5. Has this work been presented/submitted elsewhere?

No

6. Which form of presentation is preferred: Oral or Poster?

Poster

Spectral Eigenfeatures for Effective DP Matching in Fingerprint Recognition

Boris Danev and Toshio Kamei

Media and Information Research Laboratories, NEC Corporation
1753, Shimonumabe, Nakahara, Kawasaki, Kanagawa 211-8666, Japan
b-danev@zs.jp.nec.com, t-kamei@cb.jp.nec.com

Abstract. Dynamic Programming (DP) matching has been applied to solve distortion in spectral-based fingerprint recognition. However, spectral data is redundant, and its size is huge. PCA could be used to reduce the data size, but leads to loss of topographical information in projected vectors. This allows only inter-vector similarity estimations such as Euclid or Mahalanobis distances, and proves to be inadequate in presence of distortion occurring in finger sweeping with a line sensor. In this paper, we propose a novel two-step PCA to extract compact eigenfeatures amenable to DP matching. The first PCA extracts eigenfeatures of Fourier spectra from each image line. The second extracts eigenfeatures from all lines to form the feature templates. In matching, the feature templates are inversely transformed to line-by-line representations on the first PCA subspace for DP matching. Fingerprint matching experiments demonstrate the effectiveness of our proposed approach in template size reduction and accuracy improvement.

1 Introduction

Automated fingerprint identification has been used in law enforcement, and showed the identification capability of fingerprints [1, 2]. Nowadays fingerprint recognition is being widely spread in various applications, especially consumer devices such as personal computers, mobile phones, and USB keys [3]. In these devices, fingerprint sensors are embedded to authenticate the validity of users. The most popular sensor is a sweep-type line sensor because of its cost and size. The users sweep their fingers on the sensor, and fingerprint images are captured and recognized. However, current fingerprint recognition technologies are not enough to satisfy all the requirements such as accuracy, size and cost for embedded devices.

Matsumoto et al. [4] proposed fingerprint verification methods based on DP matching of spectral features. In this work, spectral features such as Linear Prediction Coefficients (LPC) were extracted from each image line. DP treats the spectral features as one-dimensional signals to solve distortion in fingerprint images. Although this approach might have the potential to satisfy the requirements for embedded devices, spectral data contains redundant information, and its size is huge. Principal Component Analysis (PCA) is effective in reducing data

size by keeping original information, and therefore it is widely used in biometrics [5–7]. PCA projects the original feature into a PCA subspace. Inter-vector distances such as Euclid or Mahalanobis distances are often used to compute similarity between projected vectors. In case of spectral features of fingerprints, PCA could be used to reduce the data size, but leads to loss of topographical information in projected vectors. Therefore, DP matching is not applicable to the projected vectors. Even if inter-vector distances were used, it is difficult to reach enough accuracy due to the presence of distortion occurring in finger sweeping with a line sensor.

Our challenge is how to introduce an efficient PCA technique into DP matching of spectral features. In this paper, we propose a novel two-step PCA to extract compact eigenfeatures amenable to DP matching. The first PCA is used to extract eigenfeatures of Fourier spectra from each image line. Then a second PCA is applied to extract the eigenfeatures from all image lines in order to form the feature templates. In matching phase, the feature templates are inversely transformed into the first PCA subspace to reconstruct line-by-line representations. Then we apply DP matching to the line-by-line representations. Furthermore, in order to enhance the second PCA, we introduce automatically generated images into the estimation of the second PCA matrix. These images are generated from the original images based on warping information provided by DP matching.

2 Spectral Eigenfeatures for DP matching

2.1 Spectral Eigenfeature Extraction Using Two-step PCA

In feature extraction, spectral eigenfeatures are extracted using two-step PCA to obtain compact features amenable to DP matching as summarized in Fig. 1(a).

First, we apply the one-dimensional Fourier transformation line by line to the normalized image $f(m, n)$ of a given fingerprint:

$$F(\omega, n) = \frac{1}{\sqrt{M}} \sum_{m=0}^{M-1} f(m, n) \exp(-2\pi i \frac{m\omega}{M}) \quad (1)$$

where $0 \leq m \leq M - 1$ (horizontal), $0 \leq n \leq N - 1$ (vertical)

In the first PCA, a projected vector \mathbf{g}_l , also called an eigenfeature, is extracted from the Fourier amplitude spectrum of each horizontal image line l using a PCA matrix W_L to obtain its principal components:

$$\mathbf{g}_l = W_L^t \mathbf{s}_l \quad (2)$$

Here, \mathbf{s}_l denotes the spectrum of the Fourier amplitude $|F(\omega, l)|$ in a vector form: $\mathbf{s}_l = [|F(1, l)| \ |F(2, l)| \ \cdots \ |F(M/2 - 1, l)|]^t$, where the DC component and redundant half of the spectrum are removed. Extraction for all lines is written as $G = W_L^t S$ in step (a) in Fig. 1 using G , an array of \mathbf{g}_l , and a matrix $S = [\mathbf{s}_0 \ \mathbf{s}_1 \ \cdots \ \mathbf{s}_{N-1}]$, also called *spectral image* in this paper.

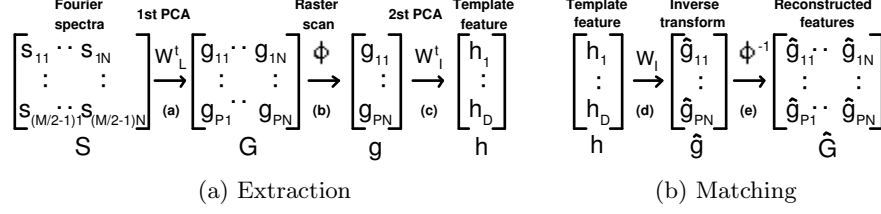


Fig. 1. A scheme for spectral eigenfeature extraction and matching

The second PCA projects a vector obtained by raster-scanning of elements in G into the PCA subspace specified by a matrix W_I to obtain the second eigenfeature \mathbf{h} as shown in step (b) and (c):

$$\mathbf{h} = W_I^t \phi(G) = W_I^t \phi(W_L^t S) \quad (3)$$

where $\phi(G)$ denotes the raster-scanning of matrix G . The second eigenfeature \mathbf{h} is enrolled as a feature template used in matching. The numbers of principal components in the first and second PCA are experimentally determined.

2.2 DP Matching Using Eigenfeatures

DP matching [4, 8] is used to match feature templates. To restore topographical information, a reference template \mathbf{h}^R and a test template \mathbf{h}^T are inversely transformed into the first PCA subspace to reconstruct line-by-line representations \hat{G}^R, \hat{G}^T using the second PCA matrix W_I :

$$\hat{G}^R = \phi^{-1}(W_I \mathbf{h}^R), \quad \hat{G}^T = \phi^{-1}(W_I \mathbf{h}^T) \quad (4)$$

where ϕ^{-1} denotes the inverse operation of raster-scanning ϕ .

As the reconstructed line-by-line representations \hat{G}^R, \hat{G}^T are approximations of the principal components of spectral features of horizontal lines, DP matching is performed in vertical direction to obtain the similarity between \hat{G}^R and \hat{G}^T .

2.3 Training of PCA matrices

PCA matrix is given as the set of κ eigenvectors in the eigenvector matrix V corresponding to the κ -highest eigenvalues in the eigenvalue problem $CV = V\Lambda$ where Λ is the eigenvalue matrix and C is the covariance matrix calculated from a training set. K normalized images $f^k(m, n)$ ($0 \leq k \leq K-1$) and their spectral images $S^k = [s_0^k \ s_1^k \ \cdots \ s_{N-1}^k]$ are used as the training set to calculate the first and second PCA matrices W_L and W_I .

Since each line is projected using the same PCA matrix in the first PCA, the matrix can be calculated from a training set $\{s_0^0, s_1^0, \dots, s_0^1, s_1^1, \dots, s_{N-1}^{K-1}\}$ consisting of all Fourier amplitude spectra s_l^k of all spectral images S^k .

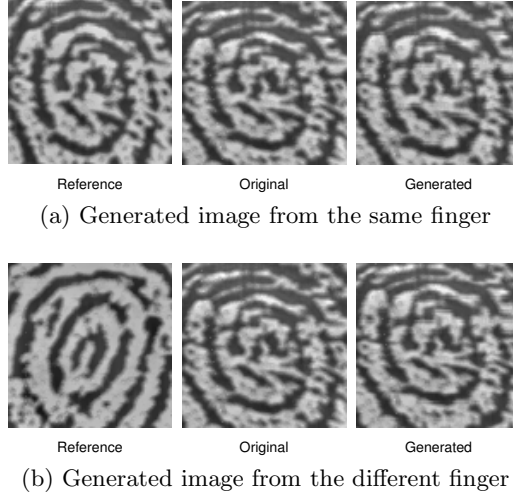


Fig. 2. Automatically generated images for training

On the other hand, the second PCA extracts the principal components of all features obtained in the first PCA. In order to increase training dataset, we introduce an automatically generated training set into the estimation of the second PCA. The generated training set $\{X^{pq}\}$ is produced by warping images in the original training set. Image X^{pq} is generated by transforming an original image f^q into the topography of the reference image f^p using warping information obtained in the DP matching between the spectral images of f^p and f^q . For example, the generated image in Fig. 2(a) has changed position of ridge bifurcation above the core, because DP matching applied on the original image with the reference image has tried to solve the deformation in the fingerprint. In the case of using a reference image from a different finger, the generated image in Fig. 2(b) looks consistently stretched compared to the original. Using the spectral images $S_{X^{pq}}$ derived from the generated images X^{pq} , the covariance matrix C_I for the second PCA is calculated:

$$C_I = \frac{1}{K^2 - 1} \sum_{p=1}^K \sum_{q=1}^K (\mathbf{x}_{pq} - \bar{\mathbf{x}})(\mathbf{x}_{pq} - \bar{\mathbf{x}})^t, \quad \mathbf{x}_{pq} = \phi(W_L^t S_{X^{pq}}) \quad (5)$$

Therefore, the number of training data becomes K^2 from K .

3 Experiments

3.1 Dataset

For experiments, we used a database of fingerprint images captured with an optical line sensor with 858 dpi resolution. The database contains images from 18 people with 8 different fingers, 3 samples per finger.

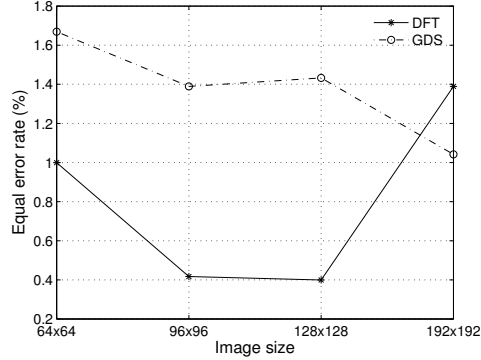


Fig. 3. Accuracy dependency of DFT and GDS on image size

We pre-processed the database in the following way: we first manually identified a reference point. A core was usually selected as such a point. In cases where core was not present, a minutia near the core region was selected. Then, a square region of size 96×96 was cropped with reference point in the middle, and pixel-value normalization was applied [9]. Furthermore, the database was divided in two datasets. The first 144 ($48 \text{ fingers} \times 3 \text{ samples}$) images formed the training set, while the other 288 images ($96 \text{ fingers} \times 3 \text{ samples}$) composed the testing set. Thus, training and testing sets were disjoint.

3.2 Spectral Features

DFT vs GDS Matsumoto *et. al* proposed Group Delay Spectrum (GDS) as DFT alternative for spectral representation of each horizontal line of image in DP matching of fingerprints [4]. We compared DFT with GDS for different image sizes using their approach. Hamming window is applied to each horizontal line in the DFT calculation. As shown in Fig. 3, GDS feature proved better than DFT feature in large size region (e.g., 192×192), and scored poorly in small size ones (e.g., 96×96). The reasons for this are probably twofold. First, as GDS uses phase information of the LPC spectrum, the short length of the one-dimensional signal is not enough to properly predict the linear coefficients. Second, larger size regions in our database had areas that include background regions. This tends to degrade discriminant capabilities of the DFT feature. Thus, we selected DFT of 96×96 cropped images in the experiments described later.

Dimensionality of Features In the Fourier spectrum s_l , we used the 47 components of the DFT amplitude spectrum. Therefore, the size of S becomes $96 \times 47 = 4512$. The first 12 principal components were extracted in the first PCA, as this experimentally proved to be enough to represent each Fourier amplitude spectrum without losing matching accuracy. The size of G becomes $96 \times 12 = 1152$.

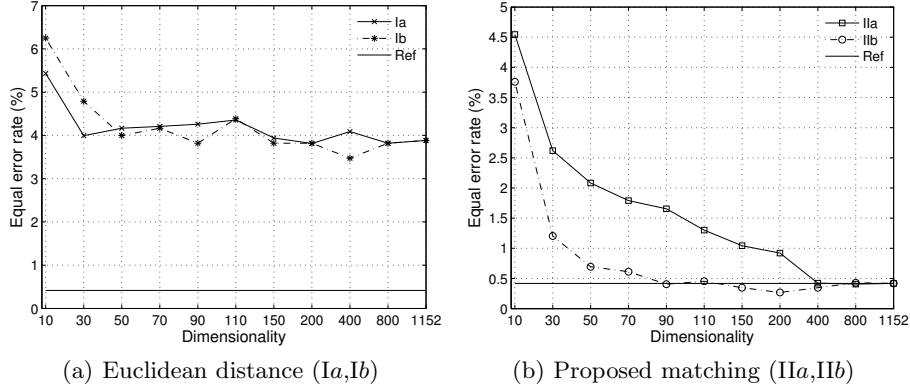


Fig. 4. Accuracy dependency on the dimensionality of feature template

Table 1. Summary of the matching results

Algorithm	Matching	Training Set	Best EER	Template Size
Ref[4]	DP matching	N/A	0.42%	4512
Ia	Euclidean distance	original images	3.81%	200
Ib	Euclidean distance	generated images	3.47%	400
IIa	Proposed matching	original images	0.42%	1152
IIb	Proposed matching	generated images	0.26%	200

3.3 Matching Experiments

We evaluated our proposed algorithm in comparison to other algorithms with respect to the feature template size and matching accuracy. The reference algorithm is the algorithm in [4] which uses DFT instead of GDS. Equal Error Rate (EER) of the reference algorithm was 0.42% at image size 96×96 . We also compared our algorithm with an algorithm based on an inter-vector matching between feature templates \mathbf{h} in Eq. 3 using the Euclidean distance. The effect of image generation in W_I computation was also investigated.

Fig. 4 shows the accuracy dependency on the dimensionality of feature template. In the figure, the suffixes a and b (e.g., Ia , Ib) denote the difference of a training set used in W_I computation; a denotes the original training images was used, and b - the generated training images was used. Although the accuracy of inter-vector matching (Ia , Ib) showed around $EER=4\%$, our proposed algorithms (IIb) reached the reference accuracy $EER=0.42\%$ with 90 dimensions, and scored the best accuracy $EER=0.26\%$ with 200 dimensions as shown in Fig. 4. The figure also shows that the automatic image generation is significantly effective in the training of IIb . Table 1 summarizes the experimental results. Template size in the table means the dimensionality of the feature template at the best EER.

The proposed algorithm is also very efficient in terms of speed and memory consumption. At the best EER, it required $200 \times 4 = 800$ bytes for template storage in floating precision, while the template size of the reference algorithm

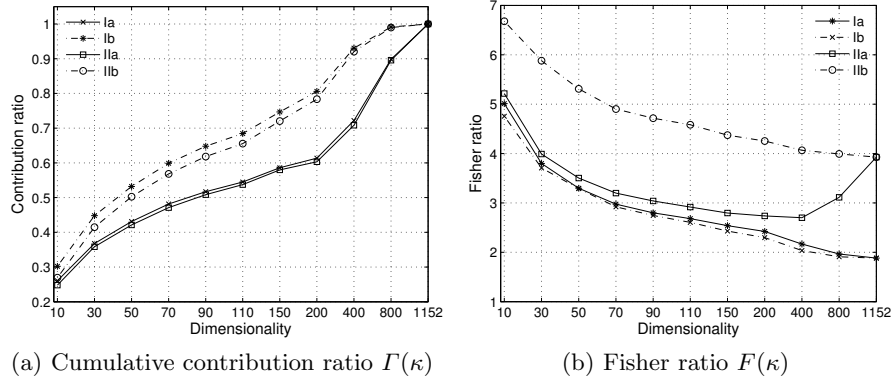


Fig. 5. Performance measures of subspaces

is 18,048 bytes. The memory size for W_L and W_I are $47 \times 12 \times 4 = 2,256$ bytes, $1152 \times 200 \times 4 = 921,600$ bytes, respectively. Matching time was about 10 ms on a Intel Pentium 4 - 3.2 GHz processor using MATLAB code.

4 Discussion

We present the results of performance measures to empirically explain the reasons behind the significant improvement achieved by the proposed algorithm. The cumulative contribution ratio and Fisher ratio of the second PCA subspace are computed as the performance measures.

The cumulative contribution ratio $\Gamma(\kappa)$ shows the extent to which κ principal components reflects the original information. It is defined as $\Gamma(\kappa) = \sum_i^{\kappa} \sigma_i^2 / \sum_i^n \sigma_i^2$, where σ_i^2 is the variance of i -th principal component, and n is the dimensionality of the original space. The variances σ_i^2 were computed using the testing set. $\Gamma(\kappa)$ with image generation (Ib, IIb) are much larger than those without image generation (Ia, IIa) as in Fig. 5(a). It means that subspaces produced with image generation effectively represent the original information. This effect is mainly obtained from the increase of training database size by the image generation.

Fisher ratio is a measure of the discriminant efficiency. For subspaces, it can be computed as $F(\kappa) = \text{tr}(S_{\kappa}^B) / \text{tr}(S_{\kappa}^W)$, where $S_{\kappa}^B, S_{\kappa}^W$ are the between and within covariance matrices of the κ -dimensional subspace. It is displayed in Fig. 5(b). Applying DP matching in subspaces (IIa, IIb) scored higher Fisher ratios which translates in their better discriminant efficiency. Furthermore, the PCA subspace calculated with generated images (IIb) exhibits the highest Fisher ratios. This effect is obtained from the warping in DP matching. In case of IIb, the distance of two features is calculated in the topography warped by DP matching, while the generated images were transformed into the similar topography. The similarity of the topography makes the image generation effective in case of IIb.

Additionally, an interesting phenomenon is observed in cases IIa, IIb at the dimensionality larger than 400. The subspaces of IIa, IIb composed of 1152 prin-

principal components are theoretically the same, which explains the same Fisher ratio achieved at the end. We expected the curves *IIa* and *IIb* on Fig. 5(b) to exhibit a smoother joining as seen in the accuracy curves in Fig. 4(b). The joining indeed occurs but it is fairly steep. A possible explanation is that although the higher components in case *IIb* represent mostly noise, in the case of *IIa* discriminant information still remains in the higher components because the PCA without image generation could not pack the relevant information into the lower subspace due to the insufficient training size and the inconsistency of topography.

5 Conclusions

We have proposed a novel way to extract spectral eigenfeatures and an effective Dynamic Programming for matching. We have also introduced an image generation technique to enhance the principal component analysis. In that context, we have proved that PCA can successfully extract very compact eigenfeatures. Learning with the image generation significantly improved the proposed scheme allowing to use only 800B for EER = 0.26% compared to the 18KB used in the reference approach with EER=0.42%.

Acknowledgment

We would like to thank the Student Exchange Office at Swiss Federal Institute of Technology (EPFL) and Prof. Takeshi Mizuno from Saitama University for the help provided in finding a post-diploma research internship in NEC Corporation.

References

1. Asai, K., Kato, Y., Hoshino, Y., Kiji, K.: Automatic fingerprint identification. Proc. Soc. of Photo-Optical Instrumentation Engineers **182** (1979) 49–56
2. Ratha, N., Bolle, R.: Automatic Fingerprint Recognition Systems. Springer, New York (2004)
3. Mainguet, J.: Biometrics for large-scale consumer products. In: Proc. Intl. Conf. on Artificial Intelligence. (2003) 310–314
4. Matsumoto, N., Sato, S., Fujiyoshi, H., Umezaki, T.: Evaluation of a fingerprint verification method based on lpc analysis. Trans. of IEE **122**(5) (2002)
5. Moghaddam, B., Pentland, A.: Probabilistic visual learning for object representation. IEEE Trans. on Pattern Anal. Machine Intell. **19**(7) (1997) 696–710
6. Kamei, T., Mizoguchi, M.: Fingerprint preselection using eigenfeatures. In: Proc. IEEE Intl. Conf. on Computer Vision and Pattern Recognition. (1998) 918–923
7. Kamei, T.: Face retrieval by an adaptive mahalanobis distance using a confidence factor. In: Proc. IEEE Intl. Conf. on Image Processing. Volume 1. (2002) I153–I156
8. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. IEEE Trans. on Acoustics, Speech and Signal Processing **26**(1) (1978) 43–49
9. Jain, A., Prabhakar, S., Hong, L.: A multichannel approach to fingerprint classification. IEEE Trans. on Pattern Anal. Machine Intell. **21**(4) (1999) 348–359