PVDC: A BINARY DESCRIPTOR USING PORE-VALLEY DISK CODE STRUCTURE FOR HIGH-RESOLUTION PARTIAL FINGERPRINT RECOGNITION

Pei-Yin Chou, Ching-Te Chiu

Institute of Information Systems and Applications, National Tsing-Hua University, Taiwan
peyin.chou@gmail.com, and ctchiu@cs.nthu.edu.tw

ABSTRACT

Plenty of pores can solve the lack of feature points problem on high-resolution partial fingerprints. Pore-based features are similar, so the neighbor ridge features are also taken into account. However, lots of feature points cause heavy computation required. We propose a binary descriptor, Pore-Valley Disk Code (PVDC), which encodes the local structure of a center pore and its neighbor valleys with an eight-section disk. The proposed descriptor is rotational invariant since the first section always aligns to the center pore orientation. Instead of recording pixel by pixel, we find that the direction and distance of intersected valleys in each section can efficiently represent the ridge structure with reduced computation. With the proposed fixed-length binary code, the matching time can be significantly reduced. The proposed method has 160x speedup compared with the state-of-the-art pore-based Sparse Representation based Direct Pore (SRDP) method with reasonable EER in HRF DBI database.

Index Terms— Pore-Valley Disk Code, High-Resolution Fingerprint, Partial Fingerprint Recognition

1. INTRODUCTION

Fingerprint recognition has been a quite popular research in computer vision. An automated fingerprint recognition system (AFRS) aims to identify the user by comparing the similarity between query fingerprint image and registered template fingerprints [1]. Recently, embedded fingerprint sensors for user authentication are widely applied on mobile devices. The insufficient panel area of sensors on mobile devices caused the captured fingerprint images may only be partial fingerprints [2]. The execution efficiency, the storage of the system and the identity accuracy are important issues to AFRS application on mobile devices.

The global characteristics such as ridge pattern [3, 4, 5] and local features such as minutiae [6, 7, 8] are mainly used for fingerprint recognition. With the progress of fingerprint sensors, the resolution of fingerprint images is higher enough (>1000dpi) to extract pores. Thus, the sufficient pores can be used to improve the recognition performance [9, 10, 11, 12, 13, 14, 15].

In previous non-pore-based methods, Minutia Cylinder Code (MCC) [6] is one of the state-of-the-art minutia-based technique on fingerprint matching. However, the computation cost of MCC is too high. Minutiae Disk Code (MDC) [7] method presents a disk structure to simply and precisely record the region information, reduces the heavy computation cost, and keeps the accuracy. However, less feature on partial fingerprints is the main problems for those minutiae-based methods.

Zhao *et al.*[9] proposed a new approach to aligning high-resolution partial fingerprint images which the floating point descriptor combines pores and valleys but it requires the heavy computation time. Zhao *et al.* developed Direct Pore (DP) [10], Sparse Representation based Direct Pore (SRDP) [11] and Tangent Distance Sparse representation based Weight RANSAC (TDSWR) [12] which use the same floating point descriptor, but SRDP and TDSWR improve the matching method. Segundo *et al.*[13] proposed a pore-based ridge reconstruction method for fingerprint recognition which use the reconstruction ridge for matching pair geometric consensus. Although those pore-based methods perform well on recognition rate, the heavy computation time is one of the problems with most of them.

In this work, we develop a pore-based binary descriptor for high-resolution partial fingerprint recognition with high efficiency and steady accuracy rate. The work of [9] first considers ridge information but have heavy execution time, while the scheme in [7] has fast speed with the binary computation. However, our proposed method considers not only relationship of pores but also ridge information with the binary operation which most of pore-based methods are floating point descriptor. The proposed PVDC descriptor can clearly record the distance and direction on the eight-section structure of the pores with reasonable Equal Error Rate(EER) and well efficiency.

2. PORE-VALLEY DISK CODE (PVDC)

Pore-Valley Disk Code (PVDC) is a binary descriptor which records the spatial and direction information of valley pixels located on the eight-section structure. The flow chart of the PVDC-based partial fingerprint recognition system is presented in Fig. 1 and three major stages are described as follows: image preprocessing, pore-valley disk code (PVDC) construction and fingerprints matching.

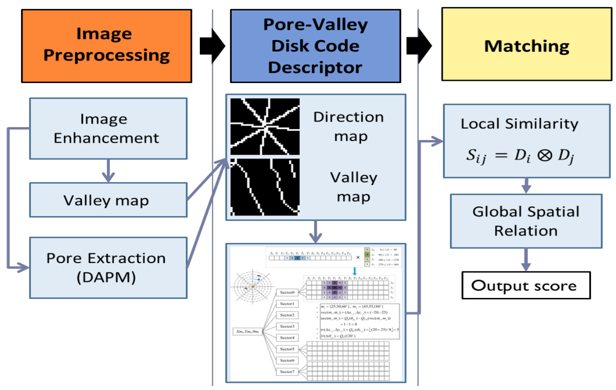

Fig. 1: The detail steps of PVDC-based partial fingerprint recognition system.

We first use [16] to enhance the fingerprint and skeleton the inverse of ridge map to get the valley map. Then the pores are extracted by implementing the pore detection algorithm described in [17].

The PVDC descriptor records the relationships between a center pore and the interested valley pixels which located on the eight-section structure with the radius is R_d. For each center pore, first is to capture the remaining valley pixels located on the eight-section structure; then the remaining valley pixels are used to construct the disk structure; finally, a descriptor is constructed after a binarization step.

2.1. Capture Remaining Valley Pixels

To reduce the computed pixels in disk structure [7], we only record the valley pixels which located on an eight-section structure. For each input center pore, we first crop the neighbor valley map with the radius is R_d. Then a eight-section direction map is constructed with the first section aligns to the center pore orientation which cause the rotation invariance of the PVDC. The directions are $\theta_j = \left\{\theta_i + \frac{360^o}{sector} \times j\right\}$, where j is the number of sector index $j = 0, 1, ..., (sector - 1)$, $sector$ is the number of sector and θ_i is the orientation of center pore. This step causes the rotation invariance of PVDC descriptor. To avoid the error caused by disconnected direction map, we apply rasterization rules on draw a full-connected line. At the end, the remaining valley pixels V_{p_i} are the intersection between valley map and direction map and is used to construct the PVDC structure for p_i.

2.2. Pore-Valley Disk Code (PVDC) Structure

Pore-Valley Disk Code (PVDC) is proposed to record the spatial and directional relationships between a center pore and the intersected valley pixels on a fixed-radius R_d multi-layer disk structure as Fig. 2. In this paper, a PVDC descriptor contains 8 sectors, 16 tracks, and 4 layers.

For each pore in the extracted pores set $p_i \in P$, a $PVDC_{p_i}$ is constructed after evaluating the relative relationships between p_i and the remaining valley pixels v_j in the remaining valley set V_{p_i}, where $j = \{v_j \in V_{p_i}, 1 \leq j \leq n_v\}$, which n_v is the number of remaining valley pixels. The disk structure represents by sector, track, and layer.

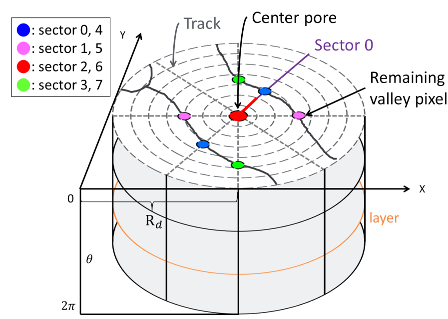

Fig. 2: An illustation of PVDC structure.

The sector inscribes the difference of direction between $vec(p_i, v_j)$ and the orientation of p_i, where $vec(p_i, v_j)$ is the vector between p_i and v_j. In this paper, the sector index is the number of direction it located on direction map.

The distance between the center pore and the neighbor valley pixel in the disk range R_d is recorded by the track T. We first calculate the Euclidean distance between the center pore and the valley pixel and quantize the distance into the range of [0,15]. The layer L records the orientation difference between the θ_{p_i} and θ_{v_j}. Then the orientation difference is quantized into a range [0,3].

A Gaussian-like smooth filter [1,2,4,2,1] is applied to the track and the layer to tolerant the captured bias from same finger, which the correspondent track and layer index get center value of the filter.

The filtered results of track and layer are the distance and direction contribution. The multiplication value of the contribution is accumulated to the correspondent sector. A real example of a PVDC structure with only one remaining pixel surrounded the center pore is shown in Fig. 3.

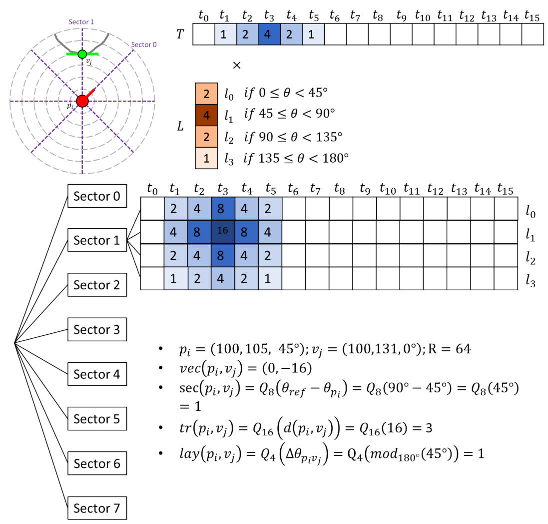

Fig. 3: An example of PVDC structure which only one remaining valley pixel around the center pore p_i. A weight filter is applied on the track and layer. The distance and direction contribution are accumulated to correspondent sector.

At the end, the accumulated disk is transferred into the binary value by a predefined threshold μ_{bin}. Fig. 4 is an real example of constructing of PVDC.

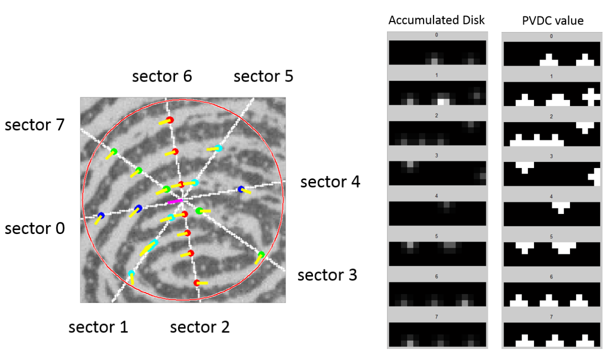

Fig. 4: An example of constructing PVDC in a real case.

2.3. Local Similarity Between Two PVDCs

To evaluate the similarity between two fingerprints (fingerprint G and H), an XOR operator is used to calculate the difference between two PVDCs. The local similarity score, $ls\left(g_i, h_j\right)$, between the descriptors D_{g_i} and D_{h_j} is defined as $ls\left(g_i, h_j\right) = \alpha \times \left(1 - \frac{|\left(D_{g_i}\right) XOR \left(D_{h_j}\right)|}{len_{PVDC}}\right)$, where len_{PVDC} is $8 \times 16 \times 4 = 512$ in this paper. A predefine constant α is used to enlarge the difference of local similarity. After each descriptor in the query image G compared with every descriptor in the reference image, a local similarity matrix $LS = \{ls\left(g_i, h_j\right), i = 1, 2, ..., n_g, j = 1, 2, ..., n_h\}$ is generated.

2.4. Geometric Consensus Examination

To evaluate the global score between two fingerprints, the main concept is iteratively checking the geometric consensus of selected correspondent pairs. Suppose fingerprint G and H have n_g and n_h pores. First, $k \times n_g$ correspondent pairs are selected by K Local Greedy (KLG) algorithm which selects the k highest local similarity score for each pore in the fingerprint G.

Given a set of correspondent candidate pairs $C_{GH} = \left\{\left(p_{g_i}, p_{h_j}, ls\left(g_i, h_j\right)\right)\right\}$, $ls\left(.\right)$ is the local similarity score between D_{g_i} and D_{h_j}. Then iteratively select two pairs, $\left(p_g^1, p_h^1\right)$ and $\left(p_g^2, p_h^2\right)$, check three conditions of geometric correlation and update the score:

$$| \, d\left(p_g^1, p_g^2\right) - d\left(p_h^1, p_h^2\right) | < Thres_d, \tag{1}$$

$$| \, \theta_g^1 - \theta_h^1 | < Thres_a, \tag{2}$$

$$| \, \theta_g^2 - \theta_h^2 | < Thres_a. \tag{3}$$

If the three conditions above are all satisfied, the scores of the two pairs are all added by 1. Otherwise, the score of the pair with less similar (i.e. the pair has lower local score) is decreased by 1, when the other pair decreases by a predefined penalty value σ.

At the end, the final matching pairs are normalized by $Score\left(G, H\right) = \frac{\left(|M_{GH}| + L_{GH}\right)^2}{\left(n_g + n_h\right)}$, where $|M_{GH}|$ is the number of final matching pairs, n_g, n_h are numbers of pores on query and reference fingerprint images and L_{GH} is normalized the mean of local similarity matrix LS by the length of descriptor.

3. EXPERIMENTAL RESULT

3.1. Experimental Environment and Setting

The proposed and compared methods are implemented in Matlab and simulated on the computer with Intel(R) Core(TM) i5-6400 CPU 2.7GHz 2.71GHz and verified on the Hong Kong Polytechnic University High-Resolution-Fingerprint (PolyU HRF) database [18] which contains two sets of high-resolution fingerprints (≥1200dpi), DBI and DBII. Both DBI and DBII database are collected from 148 fingers, each finger captured 5 images in two different sessions. The images in

DBI cover a small fingerprint area while images in DBII are full-size fingerprints.

We use the same methodology in [12] to evaluate the performance. The Equal Error Rate (EER) is the rate which means the false non-match rate (FNMR) is equal to the false match rate (FMR). The FNMR is computed by the 3700 genuine matches, while the FMR is computed by the 21756 imposter matches.

3.2. Partial Fingerprint Recognition

In this section, we compared the fingerprint recognition of our proposed method with other state-of-the-art works. Table 1 shows the EER for HRF DBI and DBII dataset with using Pore-Valley Descriptor (PVD) [9], Minutia and Pore alignment with Iterative Closest Points (MICPP) [19], DP [10], SRDP [11], TDSWR [12], pore-based ridge reconstruction [13] and the proposed approach (i.e. results for those methods are from their original paper).

The proposed PVDC descriptor is better than PVD, MICPP, and DP, but worse than SRDP, TDSWR and the method proposed by Segundo *et al.* [13]. The proposed PVDC descriptor concentrates information of a radius-R_d region into a 512-bit binary string and may cause some loss during those quantization steps. However, since SRDP, TDSWR use the same descriptor as DP and improve the matching mechanism, we convinced that our descriptor is robust enough with the performance of proposed PVDC better than DP.

Table 1: The EER (%) comparison between pore-based methods on HRF database.

Method	Descriptor Type	Descriptor Length	DBI	DBII
PVD [9]	Floating	Varying	29.5%	-
MICPP [19]	-	-	30.45%	7.83%
DP [10]	Floating	Fixed	15.42%	7.05%
SRDP [11]	Floating	Fixed	6.59%	0.97%
TDSWR [12]	Floating	Fixed	3.25%	0.53%
Segundo *et al.*s' [13]	Floating	Fixed	3.74%	0.76%
proposed PVDC	Binary	Fixed	9.75%	1.05%

3.3. Computation Time Evaluation

In addition to the recognition performance, the proposed descriptor also focuses on the efficiency. We compared it with conventional method [20], DP [10], SRDP [11], TDSWR [12] and Segundo *et al.*s' [13], MDC [7], and replaced minutiae with pores of MDC, called PDC on HRF DBI database.

To fair measure those methods, we use the same input pores and set the radius of those descriptors as 32. The average number of extracted pores and minutiae in HRF DBI database are 221 and 15 respectively. The following comparison of the execution time is based on the Matlab platform. Table 2 illustrated the comparison of average execution time. We found that our method can achieve less computation time than DP and SRDP. As TDSWR method improves SRDP on matching, the computation time is presumably similar to SRDP. Therefore, we consider that our method can have good efficiency than other compared methods.

The EER results are shown in Table 3. With using the same number of pores and same radius of the descriptors, the proposed method can achieve 9.75%, which is better than DP and SRDP. At this parameter setting, the recognition rate achieves 90.54% which the recognition rate is computed by whether the query fingerprint has the right identification result. That is the proposed method can effectively record a small region and achieve good performance with less computation time than other methods.

Table 2: The comparison of average execution time for those fingerprint matching methods on HRF DBI database with the radius is 32.

Method	Construct descriptor ($R_d = 32$) (sec/image)	Matching (1 pair of image)	Total time	Time cost
Conventional method [20]	N/A	11.07 min	664.38 sec	1703x
DP [10]	0.2 sec	3.5 sec	3.7 sec	9.48x
SRDP [11]	0.2 sec	60.5 sec	62.7 sec	160x
MDC [7]	3.8 ms	0.49 sec	0.52 sec	1.33x
PDC [7]	1.24 sec	13.89 sec	15.03 sec	39.55x
proposed PVDC	0.23 sec	0.16 sec	0.39 sec	1

Table 3: The comparison of EER(%) on HRF DBI database with the descriptor radius as 32.

Method	DP	SRDP	MDC	PDC	PVDC
EER (%)	13.08%	11.54%	32.12%	15.41%	9.75%

4. CONCLUSION

In this paper, we proposed a pore-based binary descriptor PVDC for partial fingerprint recognition. The PVDC descriptor efficiently records the valley structure surrounding the pore with reasonable performance. The proposed method can save 9x and 160x of computation time of DP and SRDP method and have the best EER during same descriptor radius.

5. REFERENCES

[1] Davide Maltoni, Dario Maio, Anil Jain, and Salil Prabhakar, *Handbook of fingerprint recognition*, Springer Science & Business Media, 2009.

[2] Biometrika inc., *A technical evaluation of fingerprint scanners*, 2001-2015.

[3] Omid Zanganeh, Bala Srinivasan, and Nandita Bhattacharjee, "Partial fingerprint matching through region-based similarity," in *Digital lmage Computing: Techniques and Applications (DlCTA), 2014 International Conference on*. IEEE, 2014, pp. 1–8.

[4] Loris Nanni and Alessandra Lumini, "Descriptors for image-based fingerprint matchers," *Expert Systems with Applications*, vol. 36, no. 10, pp. 12414–12422, 2009.

[5] Loris Nanni and Alessandra Lumini, "Local binary patterns for a hybrid fingerprint matcher," *Pattern recognition*, vol. 41, no. 11, pp. 3461–3466, 2008.

[6] Raffaele Cappelli, Matteo Ferrara, and Davide Maltoni, "Minutia cylinder-code: A new representation and matching technique for fingerprint recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 12, pp. 2128–2141, 2010.

[7] Tsai Te Chu and Ching Te Chiu, "A cost-effective minutiae disk code for fingerprint recognition and its implementation," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 981–985.

[8] Daniel Peralta, Mikel Galar, Isaac Triguero, Oscar Miguel-Hurtado, Jose M Benitez, and Francisco Herrera, "Minutiae filtering to improve both efficacy and efficiency of fingerprint matching algorithms," *Engineering Applications of Artificial Intelligence*, vol. 32, pp. 37–53, 2014.

[9] Qijun Zhao, David Zhang, Lei Zhang, and Nan Luo, "High resolution partial fingerprint alignment using pore–valley descriptors," *Pattern Recognition*, vol. 43, no. 3, pp. 1050–1061, 2010.

[10] Qijun Zhao, Lei Zhang, David Zhang, and Nan Luo, "Direct pore matching for fingerprint recognition," *Advances in Biometrics*, pp. 597–606, 2009.

[11] Feng Liu, Qijun Zhao, Lei Zhang, and David Zhang, "Fingerprint pore matching based on sparse representation," in *Pattern Recognition (ICPR), 2010 20th International Conference on*. IEEE, 2010, pp. 1630–1633.

[12] Feng Liu, Qijun Zhao, and David Zhang, "A novel hierarchical fingerprint matching approach," *Pattern Recognition*, vol. 44, no. 8, pp. 1604–1613, 2011.

[13] Mauricio Pamplona Segundo and Rubisley de Paula Lemes, "Pore-based ridge reconstruction for fingerprint recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 128–133.

[14] Rubisley de Paula Lemes, Maurlcio Pamplona Segundo, Olga RP Bellon, and Luciano Silva, "Dynamic pore filtering for keypoint detection applied to newborn authentication," in *Pattern Recognition (ICPR), 2014 22nd International Conference on*. IEEE, 2014, pp. 1698–1703.

[15] S Malathi and C Meena, "Improved partial fingerprint matching based on score level fusion using pore and sift features," in *Process Automation, Control and Computing (PACC), 2011 International Conference on*. IEEE, 2011, pp. 1–4.

[16] R. Thai, "Fingerprint image enhancement and minutiae extraction," Tech. Rep., The University of Western Australia, 2003.

[17] Qijun Zhao, David Zhang, Lei Zhang, and Nan Luo, "Adaptive fingerprint pore modeling and extraction," *Pattern Recognition*, vol. 43, no. 8, pp. 2833–2844, 2010.

[18] *PolyU HRF Database*, Available at `http://www.comp.polyu.edu.hk/biometrics/HRF/HRF_old.htm`.

[19] Anil K Jain, Yi Chen, and Meltem Demirkus, "Pores and ridges: High-resolution fingerprint matching using level 3 features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 15–27, 2007.

[20] V. K. Alilou, *Fingerprint matching: A simple approach*, [Online]. Available: `https://www.mathworks.com/matlabcentral/fileexchange/44369-fingerprint-matching--a-simple-approach`.