

A Robust Approach for Singular Point Extraction Based on Complex Polynomial Model

Jin Qi^{*◇}

^{*}*Electrical Engineering Department
University of Electronic Science & Technology of China
Chengdu, Sichuan, China
jqj@gru.edu*

Suxing Liu[◇]

[◇]*Brain and Behavior Discovery Institute
Georgia Regents University
Augusta, GA, USA
SLIU@gru.edu*

Abstract—This paper focuses on a general framework for singular point extraction from vector field. We design a new index of singular point based on complex polynomial model. We test our method in the publicly available benchmark dataset of the singular point detection competition (SPD2010). Our algorithm gets the best results and produces large margins compared to the top five algorithms which took part in the public competition. We also compare our algorithm with the state-of-the-art singular point detection algorithm (called "ZPM" method) with the benchmark. The performance of our method is much better than that of the state-of-the-art method.

Keywords-Fingerprint Singular Point; Complex Polynomial model; Vector Field;

- 1) A new invariant Angle Matching Index (AMI) is defined for singular point based on a complex polynomial model of orientation field,
- 2) Conventional convergence index filter framework is modified to give strongest evidences of presence of singular points by collecting all of the AMI information around candidate points.
- 3) We test our algorithm in publicly available benchmark dataset and compare our algorithm with state of the art methods. The experimental results show our method is much better than state of the art methods. We release our source codes to reproduce the experimental results.

I. INTRODUCTION

Singular point extraction plays a key important role in vector field analysis such as fluid field, magnetic field, gravitational field, and gradient fields from image/signal processing. The points are considered as singular points if directional fields are not continuous in these points. In this paper we emphasis the application of our singular point extraction framework in fingerprint recognition. Our method is general and can be directly extended to other applications.

Many previous works have tried to address the SPs detection and analysis problems. All of them roughly fall into two categories. The first type of approach is mainly based on the analytical properties of mathematical models estimated from fingerprint orientation field [1], [2], [3], [4]. The second tries to utilize features around singular points, such as statistical features of orientation distribution, geometrical shape features, invariant features [2], as well as appearance features.

The second type of approaches, non-model based, are sensitive to noises. The first type of methods, model based [1], [2], [3], [4], are robust to noise. We extend the model based approach and propose a general framework for singular point detection. Different from the model based approaches [1], [2], [3], [4], we define a new index for singular point based on complex polynomial model of orientation field.

The main contributions of this paper are:

II. RELATED WORK

The work most closely related to ours is model based approaches [1], [2], [3], [4]. We give a short review of these methods and point out their advantages and disadvantages.

In general, fingerprint orientation field can be fitted or approximated by mathematical models. The singular points can be detected by analyzing the analytical properties (such as critical points) of estimated mathematical models.

The authors [5] proposed a singular point extraction method based on 2D Fourier series expansions of two non-linear differential equations (FOMFE). In [5], the polynomial order is empirically set to 4 and a heuristic algorithm is applied to remove the false SPs based on the information of curl and divergence. Hence, FOMFE has limited capability to detect singular points.

In [1], authors investigated the appearance similarity between singular areas and complex filter patterns and used two kinds of complex filters to extract singular points based on the responses of different complex filters. Different from that algorithm [1], our method defines a new translation and rotation invariant feature based on complex polynomial model and a special sampling skill. Furthermore, only one filter is required to extract singular points and the specific maximum/minimum of ideal filter response (without noise) can be explicitly predicted with our model. Our method doesn't directly depends on the appearance of singular area.

Instead of estimating fingerprint orientation field model, singular points can be detected using new features obtained by deeply analyzing the mathematical model [2], [3]. A new distinct feature has been derived from zero-pole model of fingerprint orientation field in [2], [3], [4]. This feature characterizes the spacial relationship between singular points and their neighboring image pixels. This relationship is characterized by parametric curves such as straight lines in [2], [4] or circles in [3]. Then a Hough Transform (HT) based method is proposed to detect singular points [2], [3], [4].

To address the above issues (i.e., sensitive to noise and features owning limited distinguishing power) in singular point detection, we propose a new framework to generally solve the singular point extraction problem by using complex polynomial model and ridge topology analysis.

According to the protocols from the first fingerprint singular points detection competition (SPD2010) [6], we have evaluated the performance of our singular point extracting algorithm proposed here using the benchmark database and metrics suggested in SPD2010 [6], such as false alarm rate, detection rate, miss rate, percentage of correctly detected fingerprints. All of the Matlab source codes for our algorithm are available at <http://jqichina.wordpress.com/research/> to reproduce the results in this paper.

III. AMI INDEX FROM COMPLEX POLYNOMIAL MODEL

Our proposed singular point detection method needs vector field to be given. For fingerprint recognition area, The details of orientation field computation method can be found in [7]. We refer readers to the paper [7] for more specific introduction of the orientation field computation. In this paper, we use the publicly available Matlab source codes [8] of the orientation field computation algorithm mentioned above to calculate fingerprint orientation field and segment the fingerprint foreground. The Matlab source codes [8] follow the basic algorithm in [7].

A. Zero-Pole Model of Orientation Field

Zero-Pole Model is first used to model fingerprint orientation field by Sherlock and Monro [9]. Essentially, the Zero-pole model is a complex rational polynomial whose zero and pole are considered as the core and delta singular points in orientation field. The complex function $p(z)$ and the orientation $O(z)$ of pixel z in fingerprint image are given by:

$$p(z) = \sqrt{e^{2j\theta_\infty} \frac{(z - z_{c1})(z - z_{c2}) \cdots (z - z_{cm})}{(z - z_{d1})(z - z_{d2}) \cdots (z - z_{dn})}}, \quad (1)$$

$$o(z) = (\arg(p(z))) \mod \pi \quad (2)$$

where z_{ci} , z_{dj} , θ_∞ and $\arg(p(z))$ are the zeros, poles, a constant term and the argument of polynomial $p(z)$, respectively. An orientation image with a core and delta

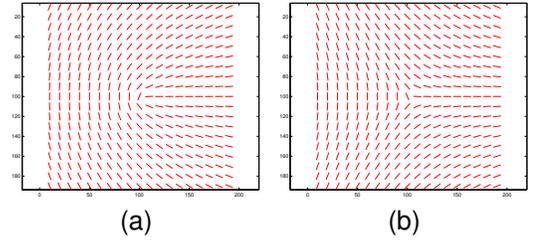


Figure 1. Zero-pole Model simulates orientation fields: (a) field with a core and (b) field with a delta.

generated by the Zero-pole Model is shown in Fig. 1 From equation (2), $o(z)$ can be represented as

$$o(z) = [o_\infty + \frac{1}{2} (\sum_{i=1}^K \arg(z - z_{ci}) - \sum_{j=1}^L \arg(z - z_{dj}))] \mod \pi. \quad (3)$$

The global topology of fingerprint image can be well described by the above zero-pole model. The model can be utilized to compute the orientation of a pixel in image and to reconstruct a good quality image from noisy image.

B. Angle Matching Index from Zero-pole Model

We derive our Angle Matching Index from the Zero-pole Model in this section. Equation (3) can be rewritten as:

$$o(z) = [o_\infty + \frac{1}{2} (\sum_{i=1}^K \arg(z - z_{ci}) - \sum_{j=1}^L \arg(z - z_{dj}))] + k \times \pi, \quad k \in \mathbb{Z}. \quad (4)$$

Hence, we have

$$2 \times o(z) = [2 \times o_\infty + (\sum_{i=1}^K \arg(z - z_{ci}) - \sum_{j=1}^L \arg(z - z_{dj}))] + 2 \times k \times \pi, \quad k \in \mathbb{Z}. \quad (5)$$

If we choose two points z_1 and z_2 in the fingerprint image, we have the following two equations:

$$2 \times o(z_1) = [2 \times o_\infty + (\sum_{i=1}^K \arg(z_1 - z_{ci}) - \sum_{j=1}^L \arg(z_1 - z_{dj}))] + 2 \times k_1 \times \pi, \quad k_1 \in \mathbb{Z}. \quad (6)$$

$$2 \times o(z_2) = [2 \times o_\infty + (\sum_{i=1}^K \arg(z_2 - z_{ci}) - \sum_{j=1}^L \arg(z_2 - z_{dj}))] + 2 \times k_2 \times \pi, \quad k_2 \in \mathbb{Z}. \quad (7)$$

Once the first equation is subtracted by the second one, we have

$$\begin{aligned}
2 \times (o(z_1) - o(z_2)) &= \left[\left(\sum_{i=1}^K \arg(z_1 - z_{ci}) - \sum_{i=1}^K \arg(z_2 - z_{ci}) \right) \right. \\
&+ \left. \sum_{j=1}^L \arg(z_2 - z_{dj}) - \sum_{j=1}^L \arg(z_1 - z_{dj}) \right] \\
&+ 2 \times (k_1 - k_2) \times \pi, \quad k_1, k_2 \in \mathbb{Z} \\
&= \sum_{i=1}^K [\arg(z_1 - z_{ci}) - \arg(z_2 - z_{ci})] + \sum_{j=1}^L [\arg(z_2 - z_{dj}) \\
&- \arg(z_1 - z_{dj})] + 2 \times k_3 \times \pi, \quad k_3 \in \mathbb{Z},
\end{aligned} \tag{8}$$

where $k_3 = k_1 - k_2$.

If we put some constraints on the positions of the points z_1 and z_2 , the equation (8) can be simplified. Suppose that points z_1 and z_2 are in the neighborhood of one fingerprint singular point and the Euclidean distance $\|z_1 - z_2\|$ between z_1 and z_2 is small. Assume without loss of generality that the point pair z_1 and z_2 are close to singular point z_{c1} and the distance between them is pretty small. The spatial distribution of the fingerprint singular points and point pair z_1 and z_2 is shown in Fig. 2. Fig. 2 shows two core points z_{c1}, z_{c2} and two delta points z_{d1}, z_{d2} . From Fig. 2, we can see that the distance from either of the points z_1 and z_2 to each of the singular points z_{c2}, z_{d1} and z_{d2} is relatively large. It can be easily concluded that the relative angle between two straight lines $l_{z_1 z_{ci}}$, connecting points z_1 and z_{ci} , $i \neq 1$ and $l_{z_2 z_{ci}}$, connecting points z_2 and z_{ci} , $i \neq 1$ or $l_{z_1 z_{dj}}$, connecting points z_1 and z_{dj} , and $l_{z_2 z_{dj}}$ connecting points z_2 and z_{dj} are pretty small. Hence we have

$$\arg(z_1 - z_{ci}) - \arg(z_2 - z_{ci}) \approx 0, \quad \forall i \neq 1, \tag{9}$$

and

$$\arg(z_2 - z_{dj}) - \arg(z_1 - z_{dj}) \approx 0, \quad \forall j \in \{1, \dots, L\}. \tag{10}$$

Substituting the above two equations into the equation (8), we get

$$\begin{aligned}
2 \times (o(z_1) - o(z_2)) &= [\arg(z_1 - z_{c1}) - \arg(z_2 - z_{c1})] \\
&+ 2 \times k_3 \times \pi, \quad k_3 \in \mathbb{Z},
\end{aligned} \tag{11}$$

To make the term k_3 in the right side of the above equation disappear, we take sin function value on both sides of the above equation. Thus the following is true

$$\sin(2 \times (o(z_1) - o(z_2))) = \sin[\arg(z_1 - z_{c1}) - \arg(z_2 - z_{c1})], \tag{12}$$

The above equation is true for the special case in Fig. 2. In general, if the point pair z_1 and z_2 is in the neighborhood

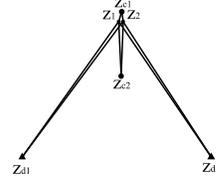


Figure 2. Spatial distribution of singular points and point pair z_1 and z_2 : core points (circle symbols) z_{c1} and z_{c2} ; delta points (triangle symbols) z_{d1} and z_{d2} ; point pair z_1 and z_2 are close to each other and in the neighborhood of singular point z_{c1} ; point pair z_1 and z_2 are far away from all of the singular points except core point z_{c1} .

of core point z_c (this case called *Case_c*) or delta point z_d (this case called *Case_d*), equation (8) can be simplified as

$$\sin(2 \times (o(z_1) - o(z_2))) = \sin[\arg(z_1 - z_c) - \arg(z_2 - z_c)], \tag{13}$$

or

$$\begin{aligned}
\sin(2 \times (o(z_1) - o(z_2))) &= \sin[\arg(z_2 - z_d) - \arg(z_1 - z_d)] \\
&= -\sin[\arg(z_1 - z_d) - \arg(z_2 - z_d)].
\end{aligned} \tag{14}$$

If the point pair z_1 and z_2 is not in the neighborhood of any singular point, the right side of equation (8) will be zero. Hence for the case where the point pair z_1 and z_2 is in the neighborhood of some non-singular point z_{ns} (this case called *Case_{ns}*), the following formula can be obtained according to equation (8)

$$\sin(2 \times (o(z_1) - o(z_2))) = 0. \tag{15}$$

For convenience, we denote $2 \times (o(z_1) - o(z_2))$ and $\arg(z_1 - z_s) - \arg(z_2 - z_s)$ by ψ and θ , respectively. Thus equation (13), (14), and (15) have the following concise forms

$$\sin \psi - \sin \theta = 0, \tag{16}$$

$$\sin \psi - \sin \theta = -2 \times \sin \theta, \tag{17}$$

and

$$\sin \psi - \sin \theta = -\sin \theta, \tag{18}$$

respectively. θ could be a small positive constant angle by specially sampling z_1 and z_2 (see more details in the next section). We call $|\sin \psi - \sin \theta|$ as "Angle Matching Index (AMI)". Apparently AMI index value is the largest (with value $|2 \times \sin \theta|$) for *Case_c* and the smallest (with value 0) for *Case_d*. Therefore, AMI index can be used to detect singular points (core and delta points). We will modify the conventional Convergence Index Filter to show how to use AMI index information around a candidate point (possibly singular or non-singular) as much as possible in the following section.

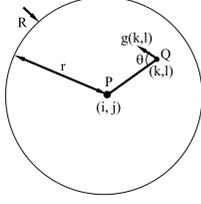


Figure 3. Convergence index filter and the angle θ : R : support region; P : center of region R ; (i, j) : coordinates of point P ; Q : arbitrary pixel; (k, l) : coordinates of pixel Q ; $g(k, l)$: gradient vector of pixel Q ; θ : relative angle between vector $g(k, l)$ and line PQ ; r : radius of support region R .

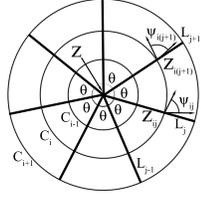


Figure 4. Extended configuration of convergence index filter and sampling pattern: z : point of interest; $C_i, i = 1, \dots, M$: concentric sampling circles; $L_j, j = 1, \dots, N$: straight lines rotating anti-clockwise every θ angle; z_{ij} : sampling point where i th circle C_i intersect with j th line L_j ; $\psi_{ij} : 2 \times (o(z_{ij}) - o(z_{i(j+1)}))$, where $o(z_{ij})$ and $o(z_{i(j+1)})$ are the fingerprint ridge orientations of pixels z_{ij} and $z_{i(j+1)}$ from the fingerprint orientation field.

C. AMI Index and Modified Convergence Index Filter

Convergence Index Filter has been widely utilized to detect rounded convex objects. Here we extend the Convergence Index Filter to detect singular points rather than convex objects. For convenience, we first give the introduction of the general Convergence Index Filter as follows.

Convergence Index Filter: As shown in Fig. 3, the filter's supporting area is a circular region R with radius r . Point P is the center of the filter. The supporting area of the filter centered around point P with coordinates (i, j) is a circular area R with a radius r . Suppose that $g(k, l)$ is the gradient vector of any point Q with coordinate (k, l) and $\theta(k, l)$ is the relative angle between gradient vector $g(k, l)$ and line \overline{PQ} connecting points P, Q . $\cos \theta(k, l)$ is defined as the **convergence index** of the gradient vector $g(k, l)$. The output $C(i, j)$ of the **Convergence Index Filter** at position (i, j) is computed as

$$C(i, j) = \frac{1}{M} \sum_{(k, l) \in R} \cos \theta(k, l), \quad (19)$$

where M denotes the total number of pixels in supporting region R . It is the same filter as the so called **COIN** filter. The convergence index at point P measure the degree to which all gradient vectors in the region R point toward the center P of the region R . It is computed using the pixels around the pixel of interest, P .

As for our case in singular point detection, we are trying to use all of the proposed AMI indices around a candidate

point. AMI index involve two angles ψ and θ which are determined by the positions of the point of interest z and the pair of sampling points which are close to each other (for example, z_1 and z_2 in Fig. 2). However, we will take a lot of sampling point pairs in the neighborhood of the point of interest z . Our sampling pattern and the extended configuration of convergence index filter are shown in Fig. 4. In Fig. 4, the circles $C_i, i = 1, \dots, M$, are concentric to each other and share the same center z . The half-lines $L_j, j = 1, \dots, N$, radiate from the point z . The orientation of the j th half-line L_j with respect to the abscissa is $2\pi j/N$, where $j = 1, \dots, N$. The sampling points z_{ij} are taken at the positions where the i th circle C_i intersects with the j th half-line L_j . We compute AMI features on all possible pairs of neighboring points z_{ij} and $z_{i(j+1)}$. Thus each sampling point z_{ij} is paired with the neighboring sampling point $z_{i(j+1)}$ and both of them lie on the same sampling circle C_i . Therefore, we assign two angles ψ_{ij} and θ_{ij} to each sampling point z_{ij} . These two angles can be computed as follows:

$$\psi_{ij} = 2 \times (o(z_{ij}) - o(z_{i(j+1)})), \quad (20)$$

and

$$\theta_{ij} = \arg(z_{ij} - z) - \arg(z_{i(j+1)} - z) = -2\pi/N = \theta, \quad (21)$$

where $o(z_{ij})$ and $o(z_{i(j+1)})$ are the ridge orientations at pixels z_{ij} and $z_{i(j+1)}$ found by fingerprint orientation field, respectively. From our special sampling pattern in Fig. 4, $\theta_{ij} = -2\pi/N$ is true. We denote the constant angle $-2\pi/N$ by θ as shown in Fig. 4

Therefore, the AMI index f_{ij} for each sampling point z_{ij} can be calculated:

$$f_{ij} = |\sin \psi_{ij} - \sin \theta_{ij}| = |\sin(2 \times (o(z_{ij}) - o(z_{i(j+1)}))) + \sin(2\pi/N)|. \quad (22)$$

This AMI index f_{ij} is similar to the convergence index value $\cos \theta(k, l)$ in equation (19). For each point of interest z as shown in Fig. 4, its AMI index value F_z is the average of all AMI indices f_{ij} in its neighborhood as follows:

$$F_z = \frac{1}{M \times N} \sum_{\substack{i=1, \dots, M \\ j=1, \dots, N}} f_{ij}. \quad (23)$$

The Average AMI value F_z is similar to the output $C(i, j)$ of the convergence index filter in equation (19).

As so far, we have extended the convergence index filter to systematically use our proposed AMI indices for singular point detection by proposing a special sampling pattern and defining new convergence indices. As the convergence index $\cos \theta(k, l)$ in equation (19) is a measure of how strongly the gradient vectors point toward the pixels of interest P in Fig. 3, our AMI index f_{ij} in equation (23) is a measure

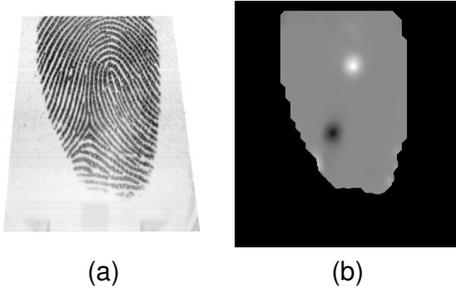


Figure 5. (a): Original fingerprint image; (b) its AMI index image.

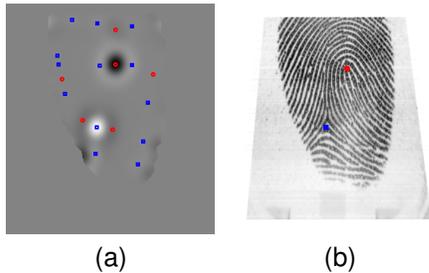


Figure 6. (a): local extrema; (b) detected singular points.

of how likely the point of interest z is a singular point in Fig. 4 (Actually, maximum for delta point, minimum for core point). An original fingerprint image and its AMI index image are shown in Fig. 5(a) and 5(b), respectively. It can be seen from Fig. 5(b) that the singular points are enhanced and non-singular points are suppressed.

D. Singular Point Detection by Thresholding

It is easy to extract singular points given AMI index image (see Fig. 5(b)) of vector field just by thresholding the AMI index image with appropriate threshold values. At the first step, we detect all of possible extrema. Then Core points are those maxima whose AMI index values are larger than $T_c = 3/2\sin(\theta) = 3/2\sin(2\pi/16) = 0.57$ and Delta points are those minima whose AMI index values are smaller than $T_d = 1/2\sin(\theta) = 1/2\sin(2\pi/16) = 0.19$. Values of thresholds T_c and T_d are empirically set in this paper. Fig. 6(a) shows our detected extrema from one orientation field. We remove false singular points by thresholding the magnitude of detected extrema and obtain the final detected singular points as shown in Fig. 6(b).

IV. EXPERIMENTAL RESULTS

In this section, some experiments will be presented. It will be shown that applying our proposed method in this paper ensures the accurate estimation of SP locations.

SPD2010 benchmark database has 500 fingerprint images with 355×390 pixel resolution captured by an optical scanner (Microsoft Fingerprint Reader - model 1033) without any restrictions on the poses of fingers. The subjects are males

and females aged from 20 to 62 years old coming from 7 countries. The fingerprint images in this dataset have a large variety in quality, type, affine transformation and nonlinear distortion.

The ground truth for the positions of core and delta points are obtained by hand according to E. R. Henry’s definition [10] of singular points.

A. Visually checking a few examples

We first visually compare the detected SP results of our algorithm with that of state of the art Zero-Pole model based method (ZPM) [3] from six types of fingerprint images(i.e. arch, tented arch, left loop, right loop, twin loop, whorl). The results are shown in supplementary material due to limited space in this paper. It can be seen that ZPM loses one true core point while our method can recover all of the true singular points from the images.

B. Quantitative performance evaluation of our method

We quantitatively evaluate the performance of our algorithm proposed in this paper by running our program on the whole SPD2010 benchmark dataset [6] consisting of 500 fingerprint images, of which 290 images and 210 images are testing dataset and training dataset, respectively. There are 240 cores and 92 deltas annotated by hand in the training set; 297 cores and 144 deltas are labeled manually in the testing set. The training database is utilized to adjust the values of the parameters in our algorithm and the state of the art method ZPM. More information on parameter setting can be found in the following sub-section.

The performance of our singular point extracting algorithm is evaluated in this paper according to the instructions from the first fingerprint singular points detection competition (SPD2010) [6]. The values of recommended metrics (such as false alarm rate, detection rate, miss rate and proportion of correctly detected fingerprints) are calculated to quantitatively measure the performance of our system. The definitions of such metrics can be found in [6].

The values of the above metrics for test data are listed in Tab. I. Our algorithm yields a substantial quantitative improvement over the other top 5 competitors in SPD2010 with this benchmark and gets the first ranking among all the algorithms. For example, compared with the best algorithm (called MagicFinger) in SPD2010 [6] our method achieves an increase of 9.38% in the percentage of correctly detected fingerprints and an increase of more than 11.60% in the core detection rate in the test dataset from SPD2010 .

We also compare our algorithm with the state of the art algorithm ZPM [3] on the same benchmark [6]. The values of metrics from our algorithm and the implemented ZPM on test dataset are listed in Tab. II. The performance of our method is much better than that of the state of the art method [3].

#	Algorithm	CD(%)	CDSP			Detection rate (%)		Miss rate (%)		False alarm rate (%)	
			#	MD	SDD	cores	deltas	cores	deltas	cores	deltas
	Ours(AMF)	38.00	240	5.11	2.61	52.00	60.00	48.00	40.00	56.00	28.00
8	MagicFinger	28.62	190	4.87	2.54	40.40	48.61	59.60	51.39	57.24	36.81
4	PintoMota	17.59	181	5.67	2.83	27.61	68.75	72.39	31.25	102.69	43.75
9	Chunfeng	17.24	136	5.63	2.55	29.29	34.03	70.71	65.97	78.79	53.47
6	Chu	15.52	119	5.58	2.57	26.60	27.78	73.40	72.22	85.36	82.64
10	PJ	12.76	96	5.86	2.36	18.18	29.17	81.82	70.83	62.96	55.56

Table I

PERFORMANCE RANKING OF DIFFERENT ALGORITHMS ON TEST DATASET. THE VALUES FOR ALL METHODS EXCEPT OURS ARE DIRECTLY FROM THE SPD2010. CD: CORRECTLY DETECTED; CDSP: CORRECT DETECTION OF SINGULAR POINTS; MD: MEAN DISTANCE; SDD: STANDARD DEVIATION OF DISTANCE.

#	Algorithm	CD(%)	CDSP			Detection rate (%)		Miss rate (%)		False alarm rate (%)	
			#	MD	SDD	cores	deltas	cores	deltas	cores	deltas
	Ours(AMF)	38.00	240	5.11	2.61	52.00	60.00	48.00	40.00	56.00	28.00
	ZPM	26.00	197	5.91	2.53	32.00	71.00	68.00	29.00	98.00	29.00

Table II

COMPARISON OF OUR ALGORITHM WITH ZPM ON TEST DATASET. CD: CORRECTLY DETECTED; CDSP: CORRECT DETECTION OF SINGULAR POINTS; MD: MEAN DISTANCE; SDD: STANDARD DEVIATION OF DISTANCE.

C. Time performance and parameter setting

The proposed SP detection algorithm in this paper is implemented in Matlab language without any optimization in programming. We evaluate the time performance of our method proposed in this paper using Intel(R) Core(TM) i7 CPU 880@3.07GHz with 64 bit windows OS. Only one core (8 cores available) is used based on single thread programming. The time used for the whole process of detection and AMI index computation part is listed in Table III for the fingerprint images with the size 355×390 pixels in the benchmark dataset. The time performance can be much improved when parallel programming skills are used. We note that the AMI index computation part takes the majority of the computing time since it uses the "nlfiler" function in Matlab which is pretty slow. The computation time can be largely reduced if AMI indices computation is implemented by using c/c++ language.

There are 6 parameters, θ , M , R , RS , T_c and T_d , in our SP detection algorithm in this paper. The values of these parameters are set by the training process in which the program run multiple times on the training database by trying different parameter values. All of the parameters and their values in this paper are listed in Table IV.

Dataset	The whole process (seconds/image)	AMI index computation (seconds/image)
SPD2010 dataset	16.4049	12.2553

Table III

THE TIME PERFORMANCE OF THE WHOLE SP DETECTION PROCESS AND AMI INDEX COMPUTATION PART IN OUR PROPOSED ALGORITHM IN THIS PAPER.

V. CONCLUSIONS

In this paper we propose a new SP detection method based on our definition of AMI index of singular points in vector fields. Our proposed approach is a general framework and

Dataset	θ	M (number of circles)	R (radius)	RS (radius step)	T_c	T_d
SPD2010 Dataset	$\pi/16$	4	5	5	0.57	0.19

Table IV
PARAMETERS AND THEIR VALUES.

can be widely used in other vector field analysis area, such as natural textures, fluid/air flow and force/electromagnetic field.

We have evaluated the performance of our singular point extracting algorithm in this paper using a number of metrics on SPD2010 dataset. The performance of our method is much better than that of the state of the art method.

ACKNOWLEDGMENT

This research has been subsidized by the scholarship for studying abroad from the electrical engineering department of University of Electronic Science & Technology of China.

REFERENCES

- [1] K. Nilsson and J. Bigun, "Localization of corresponding points in fingerprints by complex filtering," *Pattern Recognition Letters*, vol. 24, no. 13, pp. 2135 – 2144, 2003.
- [2] N. Wu and J. Zhou, "Model based algorithm for singular point detection from fingerprint images," in *Image Processing, 2004. ICIP '04. 2004 International Conference on*, vol. 2, 2004, pp. 885–888 Vol.2.
- [3] L. Fan, S. Wang, H. Wang, and T. Guo, "Singular points detection based on zero-pole model in fingerprint images," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 6, pp. 929–940, 2008.
- [4] L. Fan, S. Wang, and T. Guo, "Global and local information combined to detect singular points in fingerprint images," *Science China Information Sciences*, vol. 56, no. 6, pp. 1–13, 2013.
- [5] Y. Wang, J. Hu, and D. Phillips, "A fingerprint orientation model based on 2d fourier expansion (fomfe) and its application to singular-point detection and fingerprint indexing," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 4, pp. 573–585, 2007.
- [6] F. Magalhaes, H. P. Oliveira, and A. Campilho. Spd 2010 - fingerprint singular points detection competition database. [Online]. Available: <http://paginas.fe.up.pt/~spd2010/>
- [7] L. Hong, Y. Wan, and A. K. Jain, "Fingerprint image enhancement: Algorithm and performance evaluation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 777–789, 1998.
- [8] P. D. Kovesi, "MATLAB and Octave functions for computer vision and image processing," available from: <http://www.csse.uwa.edu.au/~pk/research/matlabfns/>.
- [9] B. Sherlock and D. Monro, "A model for interpreting fingerprint topology," *Pattern Recognition*, vol. 26, no. 7, pp. 1047 – 1055, 1993.
- [10] E. Henry, *Classification and Use of Finger Prints*. London: Routledge, 1900.