Poincare Based Singularities Detection Algorithm in fingerprint Classification

Liu Wei
School of Computer Science
Hubei University of Technology
Wuhan, Hubei 430068
China

Abstract: Fingerprint classification is an important indexing method for any large scale fingerprint recognition system or database as a method for reducing the number of fingerprints that need to be searched. Singularities are the most important and reliable features in classification. This paper describes an improved rapid singularities searching algorithm which employs delta field Poincare index. The algorithm only searches the direction field which has the larger direction changes to get the singularities. Then a singularities detection post-processing method is used to increase the accuracy. The algorithm was tested on NIST-4 database and got a good performance.

Key-Words: fingerprint, classification, singularities, Poincare index, direction field

1 Introduction

Among all the biometric indicators, fingerprint has one of the highest levels of reliability and has been extensively used. In an automatic fingerprint identification system (AFIS), the goal is to find a match for a probe fingerprint in the database of enrolled prints, possibly numbering millions. Classification is used in an AFIS to reduce the size of the search space to fingerprints of the same class before attempting exact matching. It is very important to detect singularities (core and delta) accurately and reliably for classification and matching of fingerprints. See Figure 1.

Figure 1 Sample of core and delta

A number of approaches have been developed for singularities detection in fingerprint classification. [1][2] select high curvature blocks as searching fields. In [3], minimum variance is employed to detect singularities. Nero network computing on direction field is introduced in [5]. All these approach get good performances, but these algorithms can’t justify whether the singularities exists in some position directly, and the complex post processing algorithms need to be used to get the final results.

The approach represented in this paper is based on Poincare index which can search the singularities speedy and directly. In the direction field, counterclockwise rounding a singular point along a close curve will influence the following angle changes. The angle change will be 180° while the center of the curve is core, −180° for delta and 0° for other points. We use this feature to search and detect the singularities.

The paper will be arranged as follows. Section 2 is about the Poincare index. Section 3 introduced the singularities detection. Section 4 is about some post-processing. Section 5 gives the experimental results and the performance. We draw conclusion in section 6.

2 Poincare Index

Poincare index is defined as:

\[
Poinc(i,j)=\frac{1}{2\pi} \lim_{\epsilon \to 0} \int \left( \frac{x}{\partial} \right) (x+\epsilon \cos \theta, y+\epsilon \sin \theta) d\theta
\]

(1)

\(\theta\) respresents the direction field, and the Poincare index of point \((i, j)\) is defined as:

\[
Poinc(i,j)=\frac{1}{2\pi} \lim_{\epsilon \to 0} \int \left( \frac{x}{\partial} \right) (i+\epsilon \cos \theta, j+\epsilon \sin \theta) d\theta
\]

(2)

with
\[
\frac{\partial}{\partial \theta} \sigma(i + \epsilon \cos \theta, j + \sin \theta) = \begin{cases} 
\frac{d\delta}{|d\delta|} & |d\delta| < \pi / 2 \\
\pi + \frac{d\delta}{|d\delta|} & |d\delta| \leq -\pi / 2 \\
\pi - d\delta, & \text{otherwise}
\end{cases}
\]

(3)

\[
d\delta \rightarrow \lim_{\nu \to 0} \frac{\sigma(i + \cos \nu, j + \sin \nu) - \sigma(i + \cos \theta, j + \sin \theta)}{\nu}
\]

(4)

Define \(\psi_x(\cdot)\), \(\psi_y(\cdot)\) as a close curve with \(\psi \) pixels in an image, the Poincare index will be represented as:

\[
Poincare(i, j) = \frac{1}{2\pi} \sum_{k=0}^{v} \Delta(k)
\]

(5)

where

\[
\Delta(k) = \begin{cases} 
d\delta(k), & |d\delta(k)| < \pi / 2 \\
\pi + d\delta(k), & |d\delta(k)| \leq -\pi / 2 \\
\pi - d\delta(k), & \text{otherwise}
\end{cases}
\]

(6)

\[
d\delta(k) = \sigma(\psi_x(i+1)\mod\psi, \psi_y(i+1)\mod\psi) - \sigma(\psi_x(i), \psi_y(i))
\]

(7)

3 Singularities Detection

The singularities detection method will refer the following flow.

1) compute the local direction of blocks with different size.

Typically, only \(8 \times 8\) and \(16 \times 16\) blocks will be used. And the local direction will be \(\theta_1\) and \(\theta_2\).

2) acquire the candidate field

As the angle of candidate field exists singularity will change biggish, so the difference of \(\theta_1\) and \(\theta_2\) will be higher. Use this feature, we can define:

\[
d = |\theta_1 - \theta_2|
\]

(8)

And a threshold is defined as \(T_\theta\). We will only search the blocks while \(d > T_\theta\). This will increase the detection speed.

3) compute the Poincare index.

\[
\begin{array}{ccc}
d_0 & d_7 & d_6 \\
d_1 & (i,j) & d_5 \\
d_2 & d_3 & d_4
\end{array}
\]

Figure 2. Poincare index chart

See figure 2, in close curve \(d_0, d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_0\), the Poincare index equals:

\[
Poincare(i, j) = \frac{1}{2\pi} \sum_{k=0}^{7} d_k - d_{(k+1)\mod8}
\]

(9)

4) singularities detection

In section 1, we use the feature that the angle changes is \(180^\circ, -180^\circ\) and \(0^\circ\) for core, delta and others. See figure 3. We will use this to detect the singularities.

Figure 3. angle changes and point

We can get the relationship conclusions below.

The relationship between angle changes and Poincare Index:

(a) If the angle changes equals \(180^\circ\), the Poincare index is \(\frac{1}{2}\);

(b) If the angle changes equals \(-180^\circ\), the Poincare index is \(-\frac{1}{2}\);

(c) If the angle changes equals \(0^\circ\), the Poincare index is 0;

So the points and their Poincare index is concluded as followed.

(a) Core: the Poincare index is \(\frac{1}{2}\);

(b) Delta: the Poincare index is \(-\frac{1}{2}\);

(c) Others: the Poincare index is 0.

Scanning the candidate blocks, we will get the singularities' positions and numbers after computing their Poincare index.
4 Postprocessing

The postprocessing step will eliminate the false singularities. The algorithm is:

(1) get the parameters of different radius.

In our experiment, we always choose radius 3 and 5 to confirm there’s no other singularities in the close curve. Figure 4 is the sketch map of close curve.

<table>
<thead>
<tr>
<th>$D_0$</th>
<th>$D_{15}$</th>
<th>$D_{14}$</th>
<th>$D_{13}$</th>
<th>$D_{12}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$D_1$</td>
<td>$d_0$</td>
<td>$d_7$</td>
<td>$d_6$</td>
<td>$D_{11}$</td>
</tr>
<tr>
<td>$D_2$</td>
<td>$d_1$</td>
<td>$(i, j)$</td>
<td>$d_5$</td>
<td>$D_{10}$</td>
</tr>
<tr>
<td>$D_3$</td>
<td>$d_2$</td>
<td>$d_3$</td>
<td>$d_4$</td>
<td>$D_9$</td>
</tr>
<tr>
<td>$D_4$</td>
<td>$D_5$</td>
<td>$D_6$</td>
<td>$D_7$</td>
<td>$D_8$</td>
</tr>
</tbody>
</table>

Figure 4 sketch map of close curve radius 3 and 5

And the Poincare index of close curve $d_0d_1d_2d_3d_4d_5d_6d_7d_0$ radius 3 is:

$$Poincare_1(i, j) = \frac{1}{2\pi} \sum_{k=0}^{7} [d_k - d_{(k+1)\mod 8}]$$ (10)

Similarly, the Poincare index of close curve $D_0D_1D_2\ldots D_{15}D_0$ radius 5 is:

$$Poincare_2(i, j) = \frac{1}{2\pi} \sum_{k=0}^{45} [D_k - D_{(k+1)\mod 6}]$$ (11)

Only when $Poincare_1(i, j) = Poincare_2(i, j)$, the singular point detected is effective, otherwise, the point candidate will be deleted.

5 Experiments

Figure 5 is a sample of singularities detection.

From table1, the performance is obvious satisfied for classification.

6 Conclusion

In this article, we introduce an improved singularities detection algorithm, which can searching the singularities more accurately and rapidly constration with other algorithms in section 1. The experiments performance on NIST4 is satisfied too. But the relationship formula in section 3 is too sample, and this will partly influence the accuracy. The further work will improve this.

References:


