Touchless Fingerprint Recognition System

BY MEET HARIA

UNDER THE GUIDANCE OF PROF. VIKRAM M. GADRE
Touchless Fingerprint Recognition System
Fingerprint Recognition

1. Pre-processing

2. Feature Extraction (Minutiae Extraction)

3. Matching
Finger Image Enhancement

1. Segmentation
2. Normalization
3. Estimation of Ridge Pixel Orientation
4. Estimation of Ridge Frequency
5. Gabor Filtering
6. Binarization
7. Thinning
Finger Image Segmentation

\[ V(k) = \frac{1}{W^2} \sum_{i=0}^{W-1} \sum_{j=0}^{W-1} (I(i, j) - M(k))^2 \]
Grey Level Value Normalisation

\[ N(i, j) = \begin{cases} 
M_0 + \frac{\sqrt{V_0(I(i,j)-M)^2}}{V} & \text{if } I(i, j) > M, \\
M_0 - \frac{\sqrt{V_0(I(i,j)-M)^2}}{V} & \text{otherwise}, 
\end{cases} \]
Gabor Filter

\[ G(x, y; \theta, f) = \exp \left\{ -\frac{1}{2} \left[ \frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2} \right] \right\} \cos(2\pi f x_\theta), \]

\[ x_\theta = x \cos \theta + y \sin \theta, \]

\[ y_\theta = -x \sin \theta + y \cos \theta, \]
Estimation of Ridge Pixel Orientation
Estimation of Ridge Pixel Orientation

\[
V_x(i, j) = \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} 2\partial_x(u, v)\partial_y(u, v),
\]

\[
V_y(i, j) = \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} \partial_x^2(u, v)\partial_y^2(u, v),
\]

\[
\theta(i, j) = \frac{1}{2}\tan^{-1}\frac{V_y(i, j)}{V_x(i, j)},
\]
Estimation of Ridge Pixel Orientation

\[ \Phi_x(i, j) = \cos(2\theta(i, j)), \]
\[ \Phi_y(i, j) = \sin(2\theta(i, j)), \]

\[ \Phi'_x(i, j) = \sum_{u=-\frac{w_{\Phi}}{2}}^{\frac{w_{\Phi}}{2}} \sum_{v=-\frac{w_{\Phi}}{2}}^{\frac{w_{\Phi}}{2}} G(u, v) \Phi_x(i - uw, j - vw), \]

\[ \Phi'_y(i, j) = \sum_{u=-\frac{w_{\Phi}}{2}}^{\frac{w_{\Phi}}{2}} \sum_{v=-\frac{w_{\Phi}}{2}}^{\frac{w_{\Phi}}{2}} G(u, v) \Phi_y(i - uw, j - vw), \]
Estimation of Ridge Pixel Orientation

\[ O(i, j) = \frac{1}{2} \tan^{-1} \frac{\Phi_y(i, j)}{\Phi_x(i, j)} \]

(a) Original image  (b) Orientation image
Estimation of Ridge Frequency

\[ F(i, j) = \frac{1}{S(i, j)}. \]
\[ E(i, j) = \sum_{u=-\frac{w}{2}}^{\frac{w}{2}} \sum_{v=-\frac{w}{2}}^{\frac{w}{2}} G(u, v; O(i, j), F(i, j)) N(i - u, j - v), \]
Choice of Standard Deviation

\[ \sigma_x = k_x F(i, j), \]

\[ \sigma_y = k_y F(i, j), \]
Choice of Filter Size

\[ w_x = k \cdot \sigma_x \]

\[ w_y = k \cdot \sigma_y \]
Binarization and Thinning Over Enhanced Image

(a) Enhanced image  (b) Binary image  (c) Thinned image
Binarization and Thinning Over Enhanced Image

(a) Original image  (b) Binary image  (c) Thinned image
Minutiae Extraction

<table>
<thead>
<tr>
<th>CN</th>
<th>Property</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Isolated point</td>
</tr>
<tr>
<td>1</td>
<td>Ridge ending point</td>
</tr>
<tr>
<td>2</td>
<td>Continuing ridge point</td>
</tr>
<tr>
<td>3</td>
<td>Bifurcation point</td>
</tr>
<tr>
<td>4</td>
<td>Crossing point</td>
</tr>
</tbody>
</table>
Minutiae Extraction Algorithm

\[ CN = 0.5 \sum_{i=1}^{8} |P_i - P_{i+1}|, \quad P_9 = P_1 \]
False Minutiae

(a) Spur  (b) Hole  (c) Triangle  (d) Spike
Minutia Matching

1. Fingerprint Image Registration

2. Computing Matching Score
# Android App Implementation

<table>
<thead>
<tr>
<th>Processor</th>
<th>Octa-core Qualcomm Snapdragon 616</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td>1.5 GHz</td>
</tr>
<tr>
<td>RAM</td>
<td>3 GB</td>
</tr>
<tr>
<td>Android Version</td>
<td>5.1.1</td>
</tr>
<tr>
<td>Code name</td>
<td>Lollipop</td>
</tr>
<tr>
<td>API Level</td>
<td>21</td>
</tr>
</tbody>
</table>

| Source Compatibility Java Version | 1.8        |
| Target Compatibility Java Version | 1.8        |
| Min SDK Version                  | 21         |
| Min API Level                    | 21         |

| Android Studio Version | 2.3.1 |
| Java Version          | 1.8 (Jack Enabled) |
| Compiled SDK Version  | 24    |
| Build Tools Version   | 25.0.0 |
| Target SDK Version    | 24    |
| Build Gradle Version  | 2.3.1 |
Web Server and Database

1. XAMPP
2. PHP
3. MySQL
4. phpMyAdmin
Mobile-Server Communication: HTTP URL Connection

1. HTTP URL Connection
2. JSON Object to GSON String
3. UTF-8 Encoding
4. GSON string to JSON Object
Java Socket Communication
Conclusion

1. Touchless acquisition is much more superior to touch based

2. Feasibility and Convenience due to mobile phones

3. Replaces costlier scanners

4. Feasible solution to mobile banking transaction, Criminal Identification System, mobile phone locks and much more
Future Work

1. No-tap Image Acquisition

2. Monogenic Wavelets based Pre-processing

3. Incorporation of Palmprint Biometric in the current App

4. Machine Learning Approach to Fingerprint Matching

5. Study of finger knuckles, building its identification system and thereby incorporating in the App.
References


References


Web References

http://www.sourceforge.org/blog
https://examples.javacodegeeks.com/android/core/socket-core/android-socket-example
https://www.tutorialspoint.com/sql
https://www.tutorialspoint.com/php
https://www.siteground.com/tutorials/phpmyadmin
https://blog.udemy.com/xampp-tutorial
https://www.youtube.com/watch?v=kkSG19gQamc
Thank you