
IBIMA
Publishing
mobile

Journal of Eastern Europe Research in Business and Economics

Vol. 2016 (2016), Article ID 813264, 39 minipages.
DOI:10.5171/2016.813264
www.ibimapublishing.com

Research Article

The Recognition of Fingerprints on Mobile Applications – an Android Case Study

Authors

Octavian Dospinescu[1] and Ilinca Lîsîi[2]

[1]Alexandru Ioan Cuza University, Iasi, Romania

[2] Pentalog, Iasi, Romania

Received date: 11 March 2015
Accepted date: 31 August 2015
Published date: 16 February 2016

Academic Editor: Liviu Serbanescu

Cite this Article as: Octavian Dospinescu and Ilinca Lîsîi (2016)," The Recognition of Fingerprints on Mobile Applications – an Android Case Study", Journal of Eastern Europe Research in Business and Economics , Vol. 2016 (2016), Article ID 813264, DOI: 10.5171/2016.813264

Abstract

The recognition of fingerprints has many utilities in the modern society because the fingerprint could be used as a unique id. In this article we present the main domains where the fingerprint recognition can be used, a short evolution of the fingerprint recognition process and the main steps in pre-processing the digital images. Then, we propose a new Android implementation for an application which is integrated with SourceAFIS library. Our application (BioFinger) implements the fingerprint recognition using the camera from mobile devices and we also present the obtained results and future directions.

Keywords: digital fingerprint processing, mobile integration, fingerprint recognition on Android platform

Introduction

There are many fields of application for fingerprint detection. One of them refers to biometrics in mobile voting as described by Gentles and Sankaranarayanan (2012) in a modern democratic society according to Kim and Hong (2007). Also, we consider improving the techniques for the recognition of fingerprints in mobile labs for the police activities. In the public area, there could be applications for the automation of the authentication's process in schools, hospitals and universities. In a research study by Lupu (2010), there are also concerns about car access using multimodal biometrics.

The fingerprint – evolution and algorithms of recognition

The method of fingerprint recognition has been used as a method of identifying since the nineteenth century. According to The Public Domain Review (2011), the projections and dermal folds were described for the first time in the eighteenth century, by the English botanist Nehemiah Grew. There are many algorithms and techniques for the recognition of fingerprints described by Leelambika and Rohini (2013), depending on the number of fingerprints instances and the clarity of the pictures as described by Liu and Jiang (2006). Slough History Online (2015) tell us that William James Herschel noticed that the fingerprint forms don't repeat from person to person, but he hadn't access to the necessary technical means for their final identifying and taking, an operation which is complex even today. As described by

Tredoux (2003), over half a century, Henry Faulds, an English physician, published in a journal an article that demonstrates the need for fingerprinting, using ink printing as a means of identifying offenders and villains. This approach was taken by Sir Francis Galton, naturalist and geneticist, who dedicated for ten years to study this problem. According to Findlaw (2015), he is the first to publish a paper which shows that the broad lines of the image fingertips are unique for every man, estimating that there is a possibility of 1 to 64 billion that two people have the same fingerprint. Francis Galton was able to better systematize his research, starting from the uniqueness of the personal fingerprint and establishing criteria by which it can serve to identify criminals, known as "Galton Points". His work was the beginning of the expansion and development of the technology based on fingerprint identification. Thus, it was concluded that

fingerprints can be used to identify individuals. As described by Cole (2004), in 1891, the doctor and the police chief, Juan Vucetich, creates in Argentina the first database (“vucetien” fingerprints), and he is the first who identifies, on the basis of such evidence, a criminal.

Nowadays, because of the technological progress, have appeared applications abled not only to store a large amount of data, but also to compare the fingerprints. In addition to classical methods (with white powder, fluorescent, black, special solutions, etc.), today it is often used the laser beam ("Polilight") - which, through a system of optical fibers, shows fingerprints electromagnetic spectrum –and also the "fingerprint reader" computer. Today, the fingerprinting method is used worldwide, although the fingerprint identification criteria are not so simple: modern

methodology accepts the evidence only if the research confirms that the fingerprint is based on 12 control points after a full scan. Fingerprints have become one of the most popular features used by physiological biometric technology. This popularity is due to both historical considerations and the fact that currently, they can achieve very high performance in the checking of identity. One of the immediate utility consists in the identification of the suspects in a world that, according to Oprea and Meşniţă (2007), have to fight with the global terrorism. The fingerprint is a quasi-periodic structure formed of dark paths, called papillary growths, alternating with bright trails called inter-growths. The inter-growths routes are known as papillary grooves or valleys. From the picture we can see various features of the fingerprint: the inter-growths trails and ridge lines are approximately parallel, and their orientation changes relatively slow per unit area in the

plane of the image. The points which have the most significant changes of direction are called singular points: loop and delta. Being the most visible characteristic design elements, they are called “macro details” by some authors and are used in fingerprint alignment algorithms.

The fingerprint identification points called **minutiae** refer to discontinuities that may appear in papillary ridges. There are two types of information: **termination** (ending abruptly ridges) or **fork** (which branches into two peaks).

As Ashbourn (2003) explains, the main modes of internal representation of fingerprints include: those based on main attributes of the detected details, abstracted image of the fingerprint, or the various changes of digital image’s coefficients.

The preprocessing of the fingerprints consists of:

• **Acquisition**–the fingerprints images are taken with a resolution between 250 and 625 dpi (dots per inch). Most systems use a resolution of 500dpi and the stored images are often represented using 8 bpp (bits per pixel) in 256 levels of gray.

• **Preprocessing** - higher image quality by emphasizing differences in intensity between ridge lines and inter-growths.

• **Detection of minutiae –** the automatic fingerprint identification systems use only two types of characteristic known as basic details or minutiae: ridge ending and bifurcation.

Following these steps, it is obtained the binary image, which is then used in the detection of bifurcations and ridge heads, which involves the following steps:

- thinning the ridge lines;

- details detection and classification;
    - reducing the number of false detected details.

The purpose of the fingerprint recognition algorithm is to determine whether two data come from the same fingerprint or not. In it are used various techniques that can be classified into two categories based on minutiae points or correlation. The first category involves the alignment of two sets of minutiae points and determines the total number of matches. The success of the

minutiae-based techniques is due to the accuracy use in the detection of minutiae points and also to the use of complex matching techniques in the matching process. In the second classification are compared and observed global pattern of ridges, after which it can determine the alignment of the two fingerprint. The major disadvantages of correlation-based techniques include: nonlinear distortion and the presence of "noise" in the picture. Thus, during use, minutiae-based techniques have shown to be more efficient than those based on correlation.

Due to the large variety of algorithms and sensors available on the market, standards are a key element in the process of fingerprint identification. Alonso-Fernandez et al., (2015) explain that following an intense standardization of the content and

fingerprint representation, the foundations were laid of a series of standards including: ANSI/INCITS 377-2004 Finger Pattern Based Interchange Format, ANSI/INCITS 378-2004 Finger minutiae Format for Data Interchange, ANSI/INCITS 381-2004 Finger Image-Based Data Interchange Format, ISO/IEC FCD 19794-3 Finger Pattern Based Interchange Format, ISO/IEC 19794-2 Finger minutiae Format for Data Interchange, ISO/IEC 19794-4 Finger Image Based Interchange Format.

Nowadays there are some libraries that include different types of fingerprint recognition: Android Fingerprint Scanner Software Development Kit (Bioenabletech 2015), AOSP (Slashgear 2015), AFIS (SourceAfis official documentation 2015).

The integration of fingerprint recognition systems with mobile applications

According to Pavaloaia (2013), the top activities for smartphones include the access to local information (88%) and searching for information (82%). In this article we intend to integrate the fingerprint recognition in a mobile application that could be used on an ordinary mobile phone, under the Android platform. We can choose between two main approaches: using a system based on web-services as described by Dospinescu and Percă (2013) or an autonomous mobile system. To build an autonomous system, we integrate a new library in the Android ecosystem. SourceAFIS is a library used for recognition/matching fingerprints. The essential functionality of the Automatic Identification System (AFIS) consists in comparing two fingerprints and deciding

whether they belong to the same person. SourceAFis provides fast search in a database of recorded fingerprints. It comes with an easy to use API (.NET pure and an experimental Java port) and complementary applications and tools. According to SourceAfis official documentation (2015), key features library include:

• supports fingerprint images of all common fingerprint readers;

• high search speed (10,000 fingerprints/second) in the database;

• supports and export templates according to the standard ISO/IEC 19794-2;

• average portability;

• include fingerprints visualization and analysis tool.

One of the most important steps in identifying fingerprints is purchasing it. The quality of the image obtained is the main prerequisite for continuing the recognition process. Determining the quality is carried out by the frequency analysis of the ridge and the valley, as well as the surface border.
The next step in the preprocessing is the normalization of the input image (a). The result (b) of this phase is illustrated in figure no. 1.

Please see Figure 1 in the PDF version

Please see Figure 2 in the PDF version

Please see Figure 3 in the PDF version

To be able to process fingerprints, a first step is to get this picture in high quality. SourceAFIS has no built-in method to capture fingerprints. Capturing is usually done with fingerprint readers. It is necessary to allow the takeover fingerprint readers fingerprint image, and not just specific template. After the acquisition, we obtain a fingerprint image into a standard format (eg BMP, JPEG, PNG) or a raw gray scale image. All common image formats can be used in SourceAFIS. The package SourceAFIS includes the main classes (described in AfisEngine, FingerPrint and Person Classes – 2015) used in the comparison of fingerprints:

- *AfisEngine* - Includes methods and settings for fingerprint matching engine SourceAFIS;

- *Fingerprint* - Collection of information regarding the fingerprint;
- *Person* - The collection of fingerprints belonging to a person.

The architecture of the proposed system (BioFinger) includes the following modules:

- A fingerprint reader/sensor used to collect fingerprint image and convert this into a digital format;
- a digital image processing algorithm;
- a procedure for comparison of fingerprints processed to an unknown person with those stored in a database;
- a decision procedure that uses previous comparison result in order to perform an action.

The structure of our system is presented in figure no. 4.

Please see Figure 4 in the PDF version

After we defined the general structure, we implemented the whole application using specific classes, activities, objects and methods.

The implementation of the BioFinger application

The graphical interface helps the user to collect fingerprint images and to compare them with the fingerprints from a large database. The process is very fast and it has high productivity.

Please see Figure 5 in the PDF version

The *Compare Fingerprint* button is used to call the class that implements the logic of comparing fingerprints. For the *onClick* event is called *toFingerprint Recognition ()* method. Here it is instantiated the intent that sends the two ImageView objects to the classFingerprint Activity. To compare fingerprints, the mandatory condition is to convert in ByteArray format the bitmap images in the sent list. This is achieved by the method convert BitmapToByteArray listed below.

```
public byte[] convertBitmapToByteArray(Bitmap
imgTavyIlinca) {
ByteArrayOutputStream stream = new
ByteArrayOutputStream();
Img
TavyIlinca.compress(Bitmap.CompressFormat.PNG,
100, stream);
byte[] byteArray = stream.toByteArray();
return byteArray;
}
```

The next step is the instantiation of the *Person* object, in order to get the selected fingerprints. For this, we defined a special function, *getPerson()*, which receives as parameters a value of

type *int*(used later to set id), and a parameter of type *byte[][]*, which represents the fingerprint template. The return value is of type *Person*. The logic of this method is to create an object of type *Fingerprint*, setting template by *setTemplate()*.The entire function that describes the process of creating a person and assignment of fingerprints is shown below.

```
private Person getPerson(int id, byte[][]
template) {
                Fingerprint FPlist[] = new
Fingerprint[template.length];

                for(int i=0;i<template.length;i++){

                        FPlist[i] = new
Fingerprint();

        FPlist[i].setTemplate(template[i]);
                }

                Person ps=new Person(FPlist);
                ps.setId(id);
                return ps;
        }
```

The method used to compare fingerprints is *verify()* offered by the library used in the project. This returns a float value that represents the degree of similarity between fingerprints of two people. The onCreate() method that is implemented for this process is described below.

```
@Override
protected void onCreate(Bundle
savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.tavy_layout);

lblMsg = (TextView) findViewById(R.id.txt1);

Bundle extras = getIntent().getExtras();
myList =
extras.getParcelableArrayList("imagebitmap");

myFirstImage = (ImageView)
findViewById(R.id.image1);
mySecondImage = (ImageView)
findViewById(R.id.image2);

myFirstImage.setImageBitmap(myList.get(0));
```

```
i[1] =
this.convertBitmapToByteArray(myList.g
et(1));

AfisEngineafis = new AfisEngine();
        afis.setThreshold(12);

        //getting fingerprints
         Person p1 = this.getPerson(-
1,new byte[][]{i[0]});
         Person p2 = this.getPerson(-
2,new byte[][]{i[1]});

        float matches = afis.verify(p1,
p2);
```

```
if(afis.getThreshold() < matches)
        {
       lblMsg.setTextColor(Color.GREEN);
lblMsg.setText("Fingerprints matched");
        }
       else
        {

       lblMsg.setText("Fingerprints do not
match");
       lblMsg.setTextColor(Color.RED);
        }
```

The final result consists in a decision regarding the matching between the fingerprints, as it can be seen in figure no. 6.

Please see Figure 6 in the PDF version

This result is displayed every time the user launches the matching activity from our proposed application, by comparing the initial fingerprint with the ones stored in database.

Conclusions and future directions

Our proposed application has the advantage that uses a simple smartphone which includes a photo camera and the Android operating system. The speed of processing mainly depends on two parameters: the speed of the phone processor and the size of

the database. The application could be used in different social activities and the costs are very low. One utility of the proposed application consists in quick identification of the people who participate in a big city event for promoting the city's brand (like Guinness Book records, where it is necessary to certify the identity of every participant) as described by Dospinescu N (2014). Also, the application could be deployed in the access systems of the public institutions like schools, libraries, hospitals and so on. Another utility could be the integration of fingerprint detection and recognition with the cryptography of data following the ideas offered by Airinei and Homocianu (2009); for example, the application could be extended to include a module for storing the fingerprints in a cryptographic way. In the same time, we must be careful about new threats of mobile

technologies in the age of web 3.0 as described by Popescul and Radu (2011).

As a future direction, our intention is to improve the application so it is able to implement the palm detection in order to add new capabilities to the existing architecture. In this way, the application would be useful in the process of controlling some activities by using finger or palm gestures.

References

1. AfisEngine official documentation (2015), [Online], [Retrieved February 22, 2015], http://www.sourceafis.org/blog/documentation/

2. AfisEngine Official Documentation - extension, [Online], [Retrieved January 12, 2015],FingerPrint and Person Classes, http://www.sourceafis.org/javadoc/sourceafis/simple/AfisEngine.html, http://www.sourceafis.org/javadoc/sourceafis/simple/Fingerprint.html, http://www.sourceafis.org/javadoc/

3. Airinei, D., Homocianu, D. (2009), 'An Optimized Cryptographic Way to Secure DSS Spreadsheet Reports', *Proceedings of the International Conference on Informatics in Economy*, Bucharest, 903-908

4. Alonso-Fernandez, F. et al (2015), '*Fingerprint Recognition*', [Online], [Retrieved March

2015],http://www2.hh.se/staff/josef/public/publications/alonso-fernandez09chapter.pdf, 51-90

5. Ashbourn, J. (2000), '*Biometrics: Advanced Identity Verification: The Complete Guide*', Springer-Verlag

6. Cole, S.A. (2004), '*History of Fingerprint Pattern Recognition*', Automatic Fingerprint Recognition Systems, Springer-Verlag, New York, 1-25

7. Dospinescu, N. (2014), '*The Public Relations Events in Promoting Brand Identity of the City*', Annals of Dunarea de Jos University. Fascicle I: Economics and Applied Informatics, Issue 1, 39-46

8. Dospinescu, O., Perca, M. (2013), '*Web Services in Mobile Applications*', Informatica Economica Journal, 17(2), 17-26

9. Findlaw,'*Fingerprints: The First ID*', [Online], [Retrieved February 2015], http://criminal.findlaw.com/criminal-procedure/fingerprints-the-first-id.html

10. Gentles, D., Sankaranarayanan, S. (2012), '*Application of Biometrics in Mobile Voting*', International Journal of Computer Network and Information Security (IJCNIS), 4(7), 57-68

11. http://www.bioenabletech.com/android-fingerprint-scanner-software-development-kit, [Online], [Retrieved March 06, 2015]

12. http://www.slashgear.com/android-source-code-reveals-nixed-fingerprint-support-09358810, [Online], [Retrieved March 06, 2015]

13. Kim, K., Hong, D. (2007), '*Electronic Voting System using Mobile Terminal*', World Academy of Science, Engineering and Technology, 1(8), 33-37

14. Leelambika, K.V., Rohini, A. (2013), '*Bayes Classification for the Fingerprint Retrieval*', International Journal of Advanced Research in Computer Science, 4(2), 216-220

15. Liu, M., Jiang, X., Kot, A.C. (2006), '*Fingerprint Retrieval by Complex Filter Responses*', Proceedings of 18th International Conference on Pattern Recognition, 2006, Hong Kong, 1042-1045

16. Lupu, C. (2010), '*Car Accesing Using Multimodal Biometrics*', The Annals of the "Stefan cel Mare" University of Suceava, 10, 368-377

17. Oprea, D., Mesnita, G. (2007), '*The Information Systems and the Global Terrorism*', Cyber Crime to Cyber Terrorism, Amicus Book, TheIfcai University Press, India

18. Păvăloaia, V.D. (2013), '*Methodology Approaches Regarding Classic versus Mobile Enterprise Application Development*', Informatica Economica Journal, 17(2), 59-72

19. Popescul, D., Radu, L.D. (2011), '*Mobile Threats - a Real Challenge for the Personal Information & Knowledge Management*', Proceedings of The 16th International Business

Information Management Association Conference (Innovation and Knowledge Management, A Global Competitive Advantage), June 29-30, 2011, Kuala Lumpur, Malaysia, ISBN: 978-0-9821489-5-2, 1506-1515

20. Slough History OnLine, '*William James Herschel and the discovery of fingerprinting*', [Online], [Retrieved March 01, 2015], http://www.sloughhistoryonline.org.uk/ixbin/hixclient.exe?a=query&p=slough&f=generic_theme.htm&_IXFIRST_=1&_IXMAXHITS_=1&%3Dtheme_record_id=sl-sl-williamjamesherschel

21. SourceAFIS official documentation, [Online], [Retrieved February 22, 2015], http://www.sourceafis.org/blog/documentation/

22. The Public Domain Review (2011), '*The Life and Work of Nehemiah Grew*', [Online], [Retrieved March 02, 2015], http://publicdomainreview.org/2011/03/01/the-life-and-work-of-nehemiah-grew/

23. Tredoux, G. (2003), '*Henry Faulds: the Invention of a Fingerprinter*', [Online], [Retrieved March 01, 2015], http://galton.org/fingerprints/faulds.htm