International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 3, March 2014)

# Android-Based Mobile Payment System Using 3 Factor Authentication

Saurabh Yadav[1], Pranali Patil[2], Mahesh Shinde[3], Priyanka Rane[4]

[1]*Saurabh Yadav (BE Comp S.N.D COE & RC, YEOLA)*
[2]*Pranali Patil (BE Comp S.N.D COE & RC, YEOLA)*
[3]*Mahesh Shinde (BE Comp S.N.D COE & RC, YEOLA)*
[4]*Priyanka Rane (BE Comp S.N.D COE & RC, YEOLA)*

***Abstract*-- This work is used for making transaction such as doing payment, transferring funds/money, doing online transaction via android Smartphone platforms for prominent mobile payment. This work combines a pair of Smartphone for apparent transaction, Server and paying client. These devices are featuring 3 types of authentication in this work to make a simpler and secure transaction than credit cards/atm cards/debit cards or other electronic payment cards.3 type of authentication consists of Password authentication, USIM card authentication and biometric facial authentication. This work offers simple but practical method for face recognition, eigenvectors .Eigenvectors known as eigenfaces is the approach for recognition in face classification. Research shows that proposed work have developed a near real time computer system that can locate and track a subject head and then recognize the person by comparing characteristic of the face to those of known individual.**

***Keywords*- Android, 3 ways authentication, mobile payment.**

## I. Introduction

Since the cell phone embedded with a Universal Subscriber Identity Module (USIM) card has become the most widespread device that human beings have ever created and brought along, global telecom operators are unexceptionally engaged in mobile payment service to share the ever-increasing electronic payment card market. Besides, mobile payment technology can be applied to public transportation’s electronic ticketing, vending machines, membership cards, smart poster interaction, and house electronic keys, car smart keys, official digital signatures, and soon. In this work, the user interface of the mobile payment devices is designed as easy-to-use as possible, and more importantly, the personal security must be strictly protected by 3-factor authentication feature, that is, Password authentication, USIM card authentication, and biometric facial authentication. This is shown in Figure 1

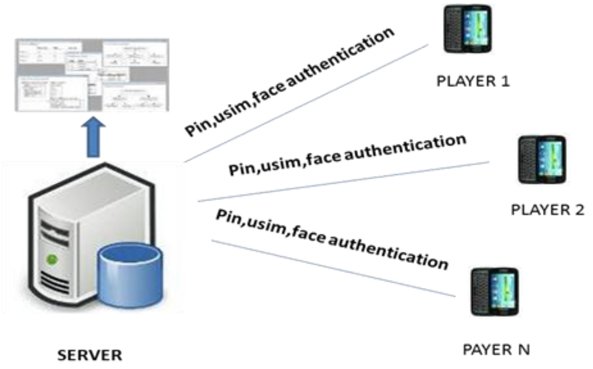

Pin,usim,face authentication
PLAYER 1
Pin,usim,face authentication
PLAYER 2
Pin,usim,face authentication
SERVER
PAYER N

**Figure 1. Mobile payment device featuring 3-type of authentication.**

In view of non invasion and cost-effectiveness, this work adopts face feature as biometric authentication rather than fingerprint feature or iris feature. While on the other hand, the security issue of Ad Hoc networking between the server and the mobile payment paying client is also critical. This work develops ports Secure Sockets Layer (SSL) protocol to put AES/DES encryption algorithm on the Ad Hoc wireless channel.

As face is primary focus of attention in social intercourse, it plays major role in conveying identity and emotion. Our task is to develop a computational model of face recognition that is fast, reasonable simple and accurate in constrained environment. There would be wide variety of problems; including criminal identification, image processing and film processing and so on .Developing a computational model of face recognition is quite difficult because faces are complex, multidimensional. We therefore are focusing our attention fully on developing pattern recognition capability that does not depend on three dimensional information or detailed geometry.

## II. Literature Survey

Current authentication systems are suffering from many weaknesses. Textual passwords are commonly used; however, users do not follow their requirements. However Users are liable to choose meaningful words from dictionaries, which make textual passwords easy to break and exposed to possibility of being attacked or harmed, or brute force attacks. Many available graphical passwords have a password space that is less than or equal to the textual password space. Smart cards or tokens can be stolen. Many biometric authentications have been put forward; however, users tend to resist using biometrics because of their intrusiveness and the effect on their privacy. Moreover, biometrics cannot be revoked.

## III. Relevance Of Work

In this work we are developing a pair of mobile payment devices, server and a paying client, on Android-based Smartphone platforms for emerging mobile payment or electronic wallet services.3 types of authentication in this work make a simpler and secure transaction than traditional credit cards or electronic payment cards.

## IV. 3 Factor Authentication Feature

In order to protect the transaction itself with the lowest risk, 3-factor authentication feature is effective and necessary. 3-factor authentication involves three different kinds of identification procedures:1) what the paying client knows (e.g., secret phrase, password, PIN code), 2)what the paying client has (e.g. token, electronic card, passport),and 3)what the paying client characterizes (e.g .behavioral or biometric feature). Only mobile payment service based on the Android based Smartphone platforms can accomplishes all requirements of 3-factor authentication easily and completely. 3-factor authentication feature has become the fundamental and common authentication standard adopted by global mobile telecom operators for promotion of mobile payment service.

### A. Password Authentication

A password is word or string of characters used for user authentication to prove their identity or validate their identity. If password is valid it will grant permission to access resource. Password should be kept secret from not allowed user or public user. Passwords are also known as passphrase. The term pass code is used when secret information is totally numerical. Such as PIN code known as personal identification number.

### B. USIM Authentication

With fast development in the mobile and wireless technology of interworking of heterogeneous networks turns into a trend and various wireless networks are getting connected with the mobile core networks through different measures. At present, in mobile communication technology, access authentication methods of various access networks are different then other and they all are based on the unique authentication algorithm in (U)SIM. In USIM authentication we add a media-independent authentication layer in USIM which outputs the uniform keys after an validating, and a key adaptation layer is designed in the terminals which transforms the output keys accordingly to meet the specific requirements of various communication. In this method, USIM is extensible in authentication algorithms and the authentication framework is independent of the communication technology. Our analysis indicates that the proposed scheme is of great advantages over the current one.

### C. Face Authentication

Eigenfaces is an alternative name given to a set of eigenvector when they are used in the computer vision problem of human recognition. The set of eigenfaces can be generated by performing a mathematical process called principle component analysis (PCA) on a large set of images depicting various human faces. Eigenfaces can also be considered a set of "standardized face features" which are derived from statistical analysis of many images of faces[7]. Human face can be considered to be the combination of these standard eigenfaces. For example, one's face may compose of the average face plus 10% from eigenface 1. 55% from eigenface 2 and even -3% of eigenface 3. Remarkably, it does not take many eigenfaces combined together to achieve a fair approximation about most faces and also, because a person's face is not recorded by a digital photograph, but instead as just a list of values much less space is taken for each person's face.

The eigenfaces that are created will appear as light and dark areas that are arranged in a particular pattern. This pattern is how different features of a face are singled out to be calculated and scored. There will be a pattern to calculated symmetry, if there is any style of facial hair, where the hairline is, or calculate the size of the mouth or nose. Other eigenfaces have specific patterns that are less easy to identify, and the image of the eigenface may look very little like a face. Performing PCA directly on the covariance matrix of the images is often computationally infeasible.

If small, say 100 × 100, greyscale images are used, each and every image is a point in a 10,000 dimensional space and the covariance matrix **S** is a matrix of 10,000 × 10,000 = $10^8$ elements. However the rank of the covariance matrix is limited by the number of training examples: if there are $N$ training examples, there will be at most $N-1$ eigenvectors with non-zero eigenvalues then we can check If the number of training examples is smaller than the dimensionality of the images, the principal element can be computed more easily as follows.

Let **T** be the matrix of preprocessed training examples, where each column contains one mean eigenvalue subtracted image [1]. The covariance matrix can be computed as $\mathbf{S} = \mathbf{T}\mathbf{T}^T$ and the eigenvector decomposition of **S** is given by

$$\mathbf{S}\mathbf{v}_i = \mathbf{T}\mathbf{T}^\mathbf{T}\mathbf{v}_i = \lambda_i \mathbf{v}_i$$

However $TT^T$ is a large matrix, and if instead we take the eigenvalue decomposition of

$$\mathbf{T}^T\mathbf{T}\mathbf{u}_i = \lambda_i \mathbf{u}_i$$

Then we notice that by pre-multiplying on both side by T, we obtain

$$\mathbf{T}^T\mathbf{T}\mathbf{u}_i = \lambda_i \mathbf{u}_i$$

Meaning that, if $u_i$ is an eigenvector of $T^TT$, then $v_i = Tu_i$ is an eigenvector of S. If we have a set of 300 images of 100×100 pixels, the matrix $T^TT$ is a 300×300 matrix, which is much more manageable than the 10,000 × 10,000 covariance matrix. Here that the resulting vectors $v_i$ are not normalized; if normalization is required it should be applied as an extra step.

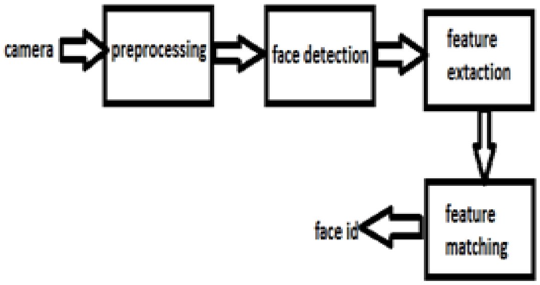


**Figure 2. Architecture of Face recognition**

User face is capture via front camera of mobile device then it is send to server where preprocessing is done on capture image. After preprocessing face is detected from preprocess image and its feature are extracted and match with the feature of user face stored on server .If the feature are matched with the feature stored on server then an OTP is generated and send to user's register mobile number via sms.

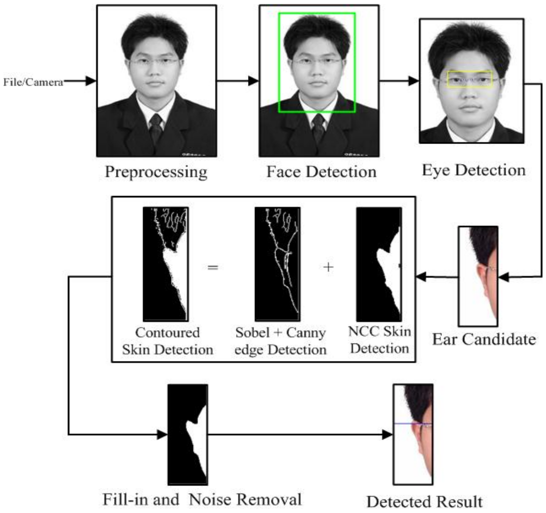


**Figure 3. Preprocessing of image**

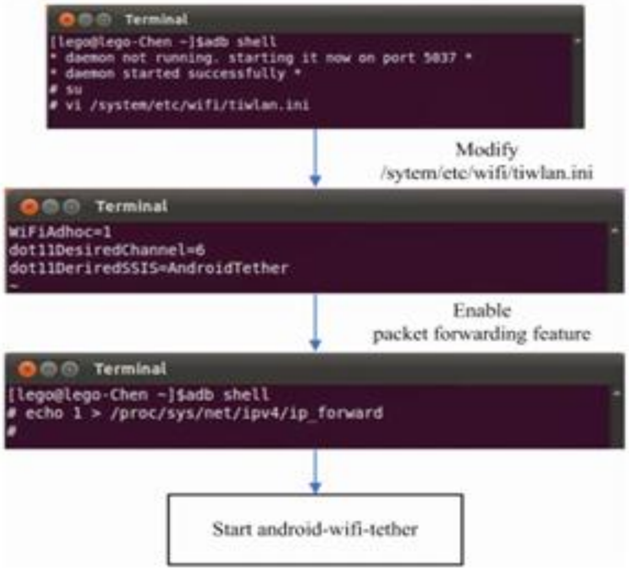
User face is capture by camera then preprocessing is done. After preprocessing face is detected then on this detected face took capture eye. Then contoured skin detection is done. Sobel and Canny operator is used on the user face. This operator are detected the edges on the user face then NCC skin detection is done. Fill-in the image and noise is removed from image then detected result.

## V. AD HOC NETWORKING CONFIGURATION

For AdHoc networking configuration, the WiFi driver of Android-based Smartphone platforms must be switched from Access Point mode to AdHoc mode throughout certificate authority, so this work adopts the boot loader-unlocked Android-based Smartphone platforms[8]. The configuration steps about Ad-Hoc-mode WiFi networking switched from Access-Point-mode WiFi networking in Android embedded.

Table 1
shows the .Recognition rate comparison between no recovery, Statistical Affine transformation [5] and proposed.

| Method | No Vertical Pose Recovery | Statistical Affine Transformation |
|---|---|---|
| Recognition rate | 74% | 86% |



```
Terminal
[lego@lego-Chen ~]$adb shell
* daemon not running. starting it now on port 5037 *
* daemon started successfully *
# su
# vi /system/etc/wifi/tiwlan.ini
```

Modify /sytem/etc/wifi/tiwlan.ini

```
Terminal
WiFiAdhoc=1
dot11DesiredChannel=6
dot11DeriredSSIS=AndroidTether
~
```

Enable packet forwarding feature

```
Terminal
[lego@lego-Chen ~]$adb shell
# echo 1 > /proc/sys/net/ipv4/ip_forward
#
```

Start android-wifi-tether

Figure 4. Configuration steps for AdHoc networking

Operating system is illustrated in Figure 5. Then, the mobile payment reader device makes good use of AdHoc networking utility, android-wifi-tether, to serve as the AdHoc gateway and create the AdHoc networking between each Android-based smart phone platform without depending upon WiFi access point.

Afterward, each Android-based mobile payment client device is regarded as a newly-joining node within this Ad Hoc network organized by the  Android-based mobile payment reader device. The interaction between a Service Set Identifier (SSID) node and a newly-joining node in AdHoc network through android-wifi-tether utility can be further optimized. In this AdHoc network, if some node receives some packet that doesn't belong to it, it will help to forward out.

### *A. I-Jetty SSL VPN Server Deployment*

To enhance the transaction security strength over the aforementioned AdHoc networking, this work takes advantage of i-Jetty open-source library and Java Server Page (JSP) programming language for SSL VPN server set-upon the mobile payment reader device. i-Jetty library is actually the light weight version of Jetty library, and It can tightly be integrated with Java applications and SSL protocol of Android embedded operating system. So building up SSL VPN server on Android-based Smartphone platforms based on i-Jetty library is time to development and time to high-performance.

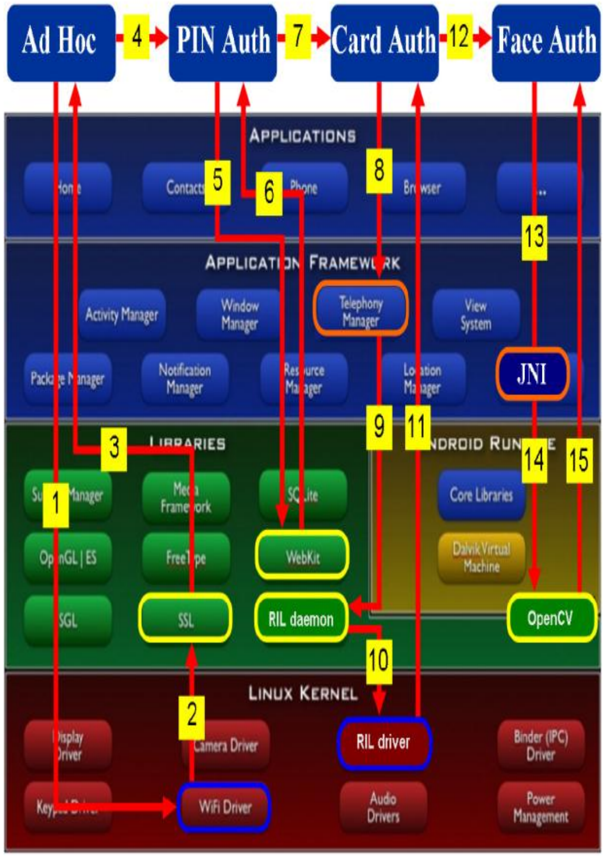


Figure 5.Implementation methodology of Android-bas ed mobile payment client device.

## VI. IMPLEMENTATION METHODOLOGY

Figure 5 specifies the overall implementation methodology of Android-based mobile payment client device featuring 3-factor authentication and virtual private Ad Hoc networking. In the device of the mobile payment client, the directional lines in Figure 5 not only represent the implementation methodology, but also indicate the execution flowchart of implemented modules interactive with Android embedded operating system architecture as shown in Figure 5, this work implements 4 applications. 1) AdHoc application to utilize Ad-Hoc-Mode WiFi driver module and Android-based SSL encryption protocol for virtual private AdHoc networking feature, 2) PIN Authentication application to employ Android-based Web Kit library and web browser for PIN code authentication feature, 3) Card Authentication application to inquire Android-based Radio Interface Layer (RIL) daemon and RIL driver module through Telephony Manager Framework for USIM card authentication, and 4) Face Authentication application to apply open-source OpenCV library through Android Java Native Interface (JNI) for facial biometric authentication.

## VII. CONCLUSION

This work, the Android-based mobile payment service using 3 factor authentications can perform well and securely. In the near future, more multifactor authentication features and virtual private networking features will keep being developed and integrated. We are going to cover all drawbacks of One Time Password in this 3 factor Authentication.

## REFERENCES

[1] Pentland, B. Moghaddam, and T. Starner, “Face Recognition Using View-Based and Modular Eigen spaces,” Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 84–91, Jun. 1994.

[2] V. Blanz and T.Vetter, “Face recognition

[3] S.Baker, I. Matthews, and J. Schneider, “Automatic Construction of active appearance models as an image coding Problem,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, pp. 1380–1384, Oct. 2004.

[4] X. Chai; S. Shan, and W. GAO, “Pose normalization for robust Face recognition based on statistical affine transformation, ”Proceedings of IEEE International Conference on Information, Communications and Signal Processing, vol. 3, pp. 1413–1417, Dec. 2003.

[5] Y.Gao, M.K. H.Leung, W. Wang, and S.C.Hui, “Fast face Identification under varying pose from a single 2-D model View,” IEEE Proceedings of Vision, Image and Signal Processing, vol. 148, pp. 248–253, Aug. 2001.

[6] M. H. Yang (2000). "Face recognition using kernel eigenfaces". Proceedings International Conference on Image Processing **1**. pp. 37–40

[7] Delac, K., Grgic, M., Liatsis, P. (2005). "Appearance-based Statistical Methods for Face Recognition". Proceedings of the 47th International Symposium ELMAR-2005 focused on Multimedia Systems and Applications, Zadar, Croatia, 08-10 June 2005, pp. 151–158

[8] Google, “Android Developers,” [Online]. Available: http://developer.android.com/index.html