IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 2, Issue 2, Apr-May, 2014
ISSN: 2320 – 8791 (Impact Factor: 1.479)
www.ijreat.org

Secure Authentication through Mixed Fingerprints

Sumathi.V[1], Thulasi.N[2], Thirunadanasikamani.k[3]

[1]PG Student, Dhanalakshmi Srinivasan College of Engineering and Technology, Mamallapuram, Chennai

[2]Associate Professor, Dhanalakshmi Srinivasan College of Engineering and Technology, Mamallapuram, Chennai

[3]Head of the Department St.Peter's Engineering College, Chennai

Abstract

In this paper,we study secure authenticated and user identification schemes based on a biometric system that can measure a user's biometric accurately . As in, the proposed approach could be used to mix the prints of the thumb and the index fingers of a single individual, or index fingers of two different individuals in order to generate a new fingerprint. Therefore, the concept of mixing fingerprints could be utilized in a multifinger We specifically focus on attacks designed to elicit information about the original bio-metric data of an individual from the stored mixed fingerprint. A few algorithms presented in the literature are discussed in this re- gard. We also examine techniques that can be used to deter or detect these attacks.

Keywords:*Mixing biometrics, phase decomposition, privacy protection, virtual Identities, Multispectral sensor.*

1. Introduction

Mixing Fingerprint systems play a crucial role in many situations where a person needs to be verified or identified with high confidence. In here, an effective fingerprint verification system is presented. It assumes that an existing reference fingerprint image must validate the identity of a person by means of a test fingerprint image acquired online and in real-time using minutiae matching[6]. The matchingsystem consists of two main blocks: (a)The first allows for the extraction of essential information from the reference image off-line, (b) The second is allowed toperforms the matching itself online. In the context offingerprints. image-level fusion has been used to combine multiple impressions of the same finger as exemplified in the following scenarios: Multispectral sensor: Rowe et al. [8] fused multiple images acquired from a multispectral fingerprint scanner into a single high quality fingerprint image. Small-area sensor: Some sensors capture only a small portion of the fingertip. Therefore, several fingerprint mosaicking techniques have been developed to stitch multiple impressions of the same finger and create a larger fingerprint. Multi-view sensor: Touchless fingerprint sensors capture mul- tiple views of a finger using several calibrated cameras [18] or asingle camera with two planar mirrors [3].

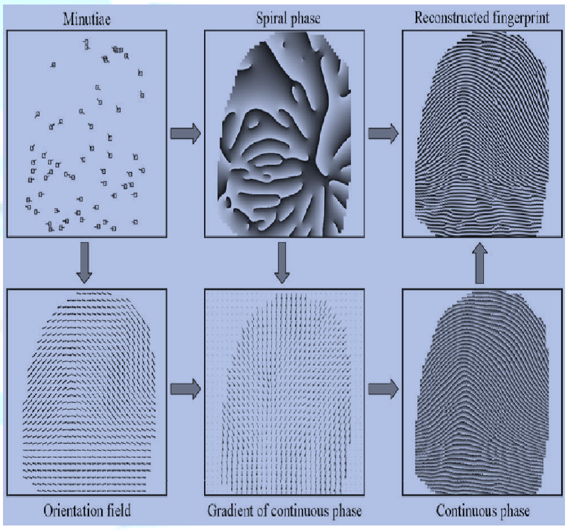

Fig.1 Phase of fingerprints

These multiple views are mosaicked together to yield a single nail-to-nail fingerprint. This work explores the possibility of mixing two different fingerprints at the image level in order to generate a new fingerprint. To mix two fingerprints, each fingerprint is decomposed into two different phase, viz., the continuous (local ridge orientation) and spiral phase (ridge ending and ridge bifurcation). These two phase are aligned to a common coordinate system. Finally, the continuous component of a one fingerprint is combined with the spiral component of a another fingerprint .There are several benefits for mixing fingerprints. For example, the proposed approach could be used to fuse images of the thumb and the index fingers of a single individual,. Traditional encryption is not sufficient for fingerprint privacy protection because decryption is required before the fingerprint matching, which exposes the fingerprint to the attacker .The information is obtained from the reference image by filtering and careful minutiae extraction procedures. The fingerprint identification is based on triangular matching to cope with the strong deformation of fingerprint images due to static friction or finger rolling. The plain whorl has one or more ridges that make a complete spiral. There are two Deltas, and if a line is drawn between them, at least one ridge in the inner pattern is touched or cut by the line .The central pocket loop whorl has one or more ridges that make a complete circle. There are two deltas, and if a line is drawn between them, no ridges in the inner pattern are touched or cut by the line. Traditional encryption is not sufficient for fingerprint privacy protection because decryption is required before the fingerprint matching, which exposes the fingerprint to the attacker

2. Working Principles

Mixed Fingerprint technique to provide more security for access the application. A combined minutiae template is generated based on Minutiae position. Orientation, Reference points detected from both fingerprints.. The user has to provide Two Different Finger prints while registration process and these finger prints are merged to form Mixed Fingerprint. Then During the login process, the user have to provide their both the finger prints and it will mixed again and compared with the original image. If the fingerprint is valid then the user is allowed to access the application. The authentication ,proposed in this paper ,is able to achieve both strong encryption-based security as well as accuracy of a powerful classifiers such as support vector machines (SVMs) and neural networks. While the proposed approach has similarities to the blind vision scheme for image retrieval ,it is far more efficient for the verification task. By to generating the One time password and send it to the users mobile phone and users E-mail Id respectively will increase the level of security .The matched templates [5] are stored into database with the help of server .Based on the SHA algorithms hash values are generated for each mixed fingerprint.

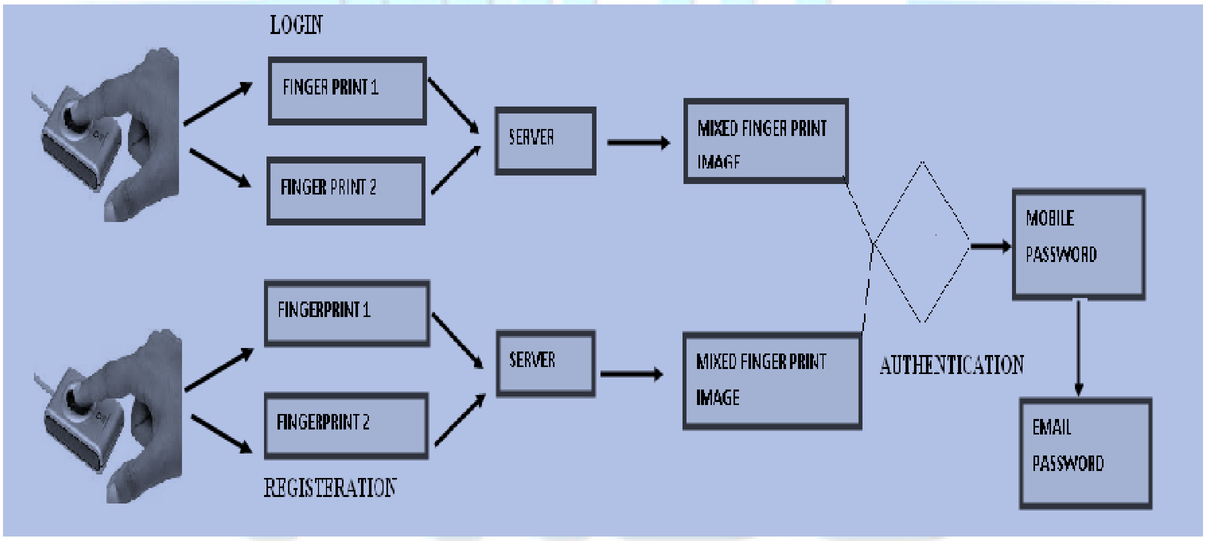

Fig. 2 Working process of mixing fingerprint

2.1 Workload model

This working models of mixing fingerprint having the following five models of process.viz.,user registration,data storage,mixed fingerprint,fingerprint verification,one time password generation.

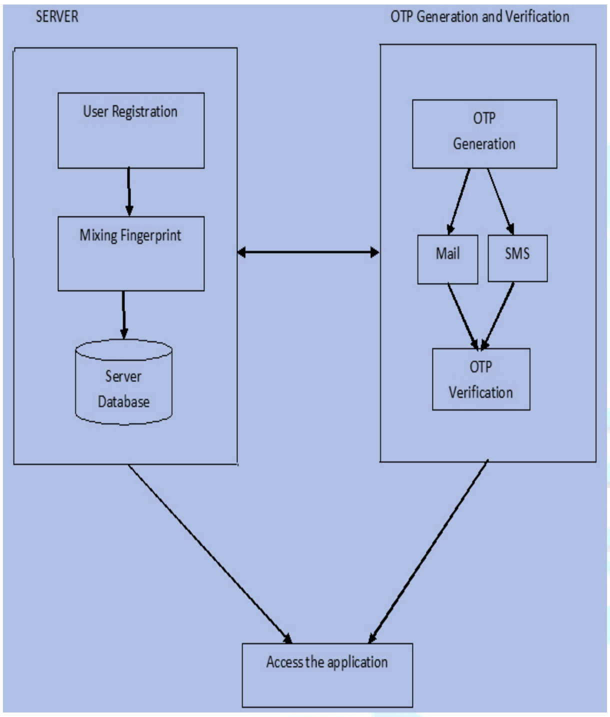

Fig.3 The proposed approach for mixed fingerprint

A.User Registration

In this module, we are going to create a User application by which the User is allowed to access the data from the Server. Here first the User wants to create an account and then only they are allowed to access the application. Once the User creates an account, they are allowed to login into their account to access the application. Based on the User's request, the Server will respond to the User. After receive a user request, then the User allowed to provide their two different fingerprint and these captured fingerprint images are mixed and stored in the database of the Server.

B. Data Storage

The Server will monitor the entire User's information in their database and verify it. Also the Server will store the entire User's information in their database. And the Server have do During the Login phase, each user is requested to enter the Sensitive information like User Id, Password. This information will be passed to the Server and the Server will verify the mixed fingerprint. If it is valid then the User is allowed tofor next level of verification.

C. Mixed Fingerprint

During the User registration Phase, the User is requested to fill all the Server requested information. Once filled, the User have to provide their two different fingerprint and these captured fingerprint images are mixed and stored in the database with of the Server. In here, threshold values are generated for each mixed fingerprint then it will be stored into separate database.

D. Fingerprint Verification

During the Login phase, each user is requested to enter the Sensitive information like User Id, Password. Once this information are verified then the User is requested to provide their two fingerprint that they have provided during the registration phase. This information will be passed to the Server and the Server will verify the mixed fingerprint. If it is valid then the User is allowed to for next level of verification.

E. One Time Password Generation

Once the Fingerprint verification is valid, An One Time Password will be generated and send to the User's Mobile Number. To generate an One time Password to implement a Secure Random Number Generation Algorithm. To send the generated One time Password to the User's Mobile Number, a JSMS jar file.Which is used to the transmit the SMS via specified port. Once the User Enters their mobile password correctly, another one time password will be generated and send to the Email address. For this process will need to Internet connection to send the password to the User Specified Email address during the registration phase. If the User provides the Email password correctly, then they are allowed to access the application.It has following operations,

(i) OTP Generator

This OTP is based on the very popular algorithm HMAC SHA. The HMAC SHA is an algorithm generally used to perform authentication by challenge response. It is not an encryption algorithm but a hashing algorithm that transforms a set of bytes to another set of bytes. This algorithm is not reversible which means that you cannot use the result to go back to the source.A HMAC SHA uses a key to transform an input array of bytes. The key is the secret that must never be accessible to a hacker and the input is the challenge. This means that OTP is a challenge response authentication.The secret key must be 20 bytes at least; the challenge is usually a counter of 8 bytes which leaves quite some time before the value is exhausted. The algorithm takes the 20 bytes key and the 8 bytes counter to create a 8 digits number. This means that there will obviously be duplicates during the life time of the OTP generator but this doesn't matter as no duplicate can occur consecutively and an OTP is only valid for a couple of minutes. There are few reasons why this is a very strong method.

- The key is 20 digits
- A password is a couple counter/password, only valid once and a very short time
- The algorithm that generates each password is not reversible
- With an OTP token, the key is hardware protected
- If the OTP is received on your phone, the key always stays at the server

Those few characteristics make the OTP a strong authentication protocol. The weakness in an authentication is usually the human factor. It is difficult to remember many complex passwords, so users often use the same one all across the internet and not really a strong one. With an OTP, you don't have to remember a password, the most you would have to remember would be PIN code (4 to 8 digits) if the OTP token is PIN protected. In the case of an OTP sent by a mobile phone, it is protected by your phone security. A PIN is short but you can't generally try it more than 3 times before the token is locked.The weakness of an OTP if there is one, is the media used to generate or receive the OTP. If the user loses it, then the authentication could be compromised. A possible solution would be to protect this device with a biometric credential, making it virtually totally safe.

(ii) The OTP Server and Authentication Protocol

The following characteristic is very important in term of security. An OTP depends on 2 parameters(a)A secret key, (b) A counter. Even if a hacker intercepts millions of OTP the algorithm is not reversible which means that even if you know the key you can't go back to the counter that was used to generate the OTP. So without the key and the counter, it is virtually impossible even with millions of OTP to find a pattern to guess the key and the current counter value .Like many security protocols, the strength of the OTP is given by the quality of the cryptography algorithm used, in this case HMACSHA1 which is a proven challenge response algorithm. An other HMAC algorithm can be used in place of HMACSHA as encryption algorithm have to become stronger when CPU power is increasing. This can be done by increasing the size of the key or by redesigning the algorithm itself. OTP are usually used to perform authentication or to verify a transaction with a credit card. In the case of a transaction an OTP is sent to the mobile phone of the user, for an authentication if is possible to use either a secure token or to request an OTP to be send to the user phone.

(iii) Using an OTP sent to a phone

This is usually the authentication method used when a transaction is verified with an OTP. The bank system sends you an OTP and you then have few minutes to enter this OTP. This mechanism doesn't need any synchronization process as the OTP is originally generated by the server and send to a third party device. The server expects that you type the correct OTP within generally 2mns. If you fail to do it, you just ask a new OTP and then enter it within the given time. When a system supports both authentication methods, it means that the back-end has 2 different keys and counters; one pair for the OTP token and one pair for the OTP transmitted by SMS.

(iv) Using an OTP token

The original product I worked on when we implemented one of the first versions of the OTP in a Javacard was using an OTP token with a screen or a mobile phone with a card applet to generate the OTP. In this model both the server and the authentication token have to generate an OTP that must be synchronized.The process is the following: The user generates an OTP with his token, type it and press OK. The server receives the OTP generated by the token, it increments the counter and generates a new OTP.

3. Experimental Result

The minutiae template is assumed to be a sequence of (r,c, θ) valuesrepresenting the location and orientation of component fingerprint minutiae. The STG begins by generating a fixed number of synthetic templates each comprising of randomly generated minutiae points.These templates are compared against the target template in the database (via the matcher) and the synthetic template resulting in the best match score is retained.The retained template is then modified iteratively via the following four operations: (i) the r, c and θ values of an existing minutia are perturbed, (ii) an existing minutia is replaced with a new minutia, (iii) a new minu- tia is added to the template, and (iv) an existing minutia is deleted. The modified template (Ti j) is compared against the target template and the match score (S(Di,Ti j)) computed. This process, viz., modifying the current synthetic template and comparing it against the target template, is repeated until the match score exceeds a pre-determined threshold. The authors used this scheme to break into 160 fingerprint accounts; their algorithm required only 271 iterations, on an average, to exceed the matching threshold for each one of those 160 accounts.other three phase just make the fingerprint appear realistic.

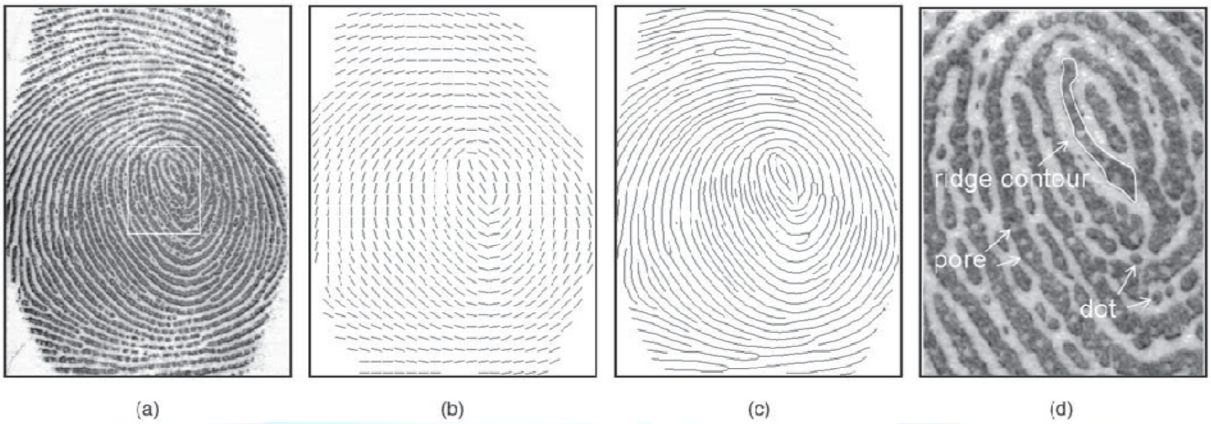

Fig. 4 Features at three levels in a fingerprint. (a) Grayscale image , (b) Level 1 feature (orientation field), (c) Level 2 feature (ridge skeleton), and (d) Level 3 features (ridge contour, pore, and dot).

3.1 fingerprint Representation

Larkin and Fletcher [25] proposed representing a finger- print image as a 2D amplitude and frequency modulated (AM-FM) signal:

$$I(x,y)=a(x,y)+b(x,y)cos(w(x,y))+n(x,y) \qquad \text{------(1)}$$

which is composed of four phase: ‘a(x,y)’ is the intensity offset, ‘b(x,y)’ is the amplitude , ‘w(x,y)’ is the phase, and ‘n(x,y)’ is noise.Here, we are only interested in the phasen(x,y). since ridges and minutiae are totally determined by the phase; the other three phase just make the fingerprint appear realistic.Therefore, an ideal fingerprint can be represented as a 2D:

$$I(x,y)=cos(w(x,y)) \qquad \text{------(2)}$$

To obtain the phase w(x,y), the following four steps are performed: (i) orientation field reconstruction,(ii) estimation of gradient of continuous phase, (iii) continuous phase reconstruction, and (iv) combination of the spiral phase and the continuous phase.which is composed of four phase: ‘a(x,y)’ is the intensity offset, ‘b(x,y)’ is the amplitude , ‘w(x,y)’ is the phase, and ‘n(x,y)’ is noise.Here, we are only interested in the phasen(x,y). since ridges and minutiae are totally determined by the phase; thefrequency. According to the Helmholtz Decomposition Theorem [26], the phase can be uniquely decomposed into two parts:the continuous phase and the spiral phase.Fig. 6 shows that the local ridge orientation in the neighborhood of the spiral is slightly changed after the spiral is added. In fact, the addition of the spiral also affects the local ridge orientation in the entire image. A minutia emerges after adding a spiral to the continuous phase. This phenomenon is very common in the area close to the delta of fingerprints and in the funnel area of palmprints.A minutia emerges after adding a spiral to the continuous phase. Assume a positive spiral is added to the continuous phase shown in Fig. 6, which is a plane slanted along the y direction. The gradient of the continuous phase is a constant vector field.In fact, the addition of the spiral also affects the local ridge orientation in the entire image.

4. Conclusion and future work

Novel mixing fingerprint scheme has been proposed which is based on converting the minutiae representation to the phase representation . The phase is composed of the continuous phase and the spiral phase. A reconstructed fingerprintis obtained by ,reconstructing the orientation field, reconstructing the continuous phase, and combining the continuous phase with the spiral phase. . The experimental results show that the mixing image is very consistent with the original fingerprint and that there is a high chance of deceiving a state-of-the-art commercial fingerprint matching system There constructed mixing fingerprints still contain a few problems like hackers knowledge about the password To overcome this, to introduce mixed fingerprint & OTP (One Time Password) mechanism for protecting the sensitive application. To reduce the risk of attacks using mixing fingerprints, robust fingerprint template security [23] and spoof detection techniques should be developed.

References

[1]A.Ross, K.Nandakumar, and A.Jain, Handbook of Multibiometrics. New York: Springer-Verlag, 2006.

[2] D. Maltoni, D. Maio, A. Jain, and S. Prabhakar, Handbook of Finger- print Recognition. NewYork:Springer-Verlag,2009.

[3] A. Othman and A. Ross, “Mixing fingerprints for generating virtual identities,” in Proc. IEEE Int. Workshop Information Forensics and Security (WIFS), Foz do Iguacu, Brazil, Nov./Dec. 2011.

[4] R. Cappelli, “Sfinge: Synthetic fingerprint generator,” in Proc. Int. Workshop Modeling and Simulation in Biometric Technology, 2004.

[5] A. Jain, A. Ross, and U. Uludag, “Biometric template security: Challenges and solutions,” in Proc. Eur. Signal Processing Conf. (EUSIPCO), 2005.

[6] G. I. Davida, Y. Frankel, and B. J. Matt, “On enabling secure appli- cations through off-line biometric identification,”inProc.IEEESymp. Security and Privacy, 1998.

[7] N. Ratha, J. Connell, and R. Bolle, “Enhancing security and privacy in biometrics-based authentication systems,” IBM Syst. J., vol. 40, no. 3, pp.

[8]A.Ross,J.Shah,andA.Jain,“Fromtemplatetoimage:Reconstructing fingerprints from minutiae points,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 4, pp. 544–560, Apr. 2007.

[9] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni, “Fingerprint image reconstruction from standard templates,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 9, pp. 1489–1503, Sep. 2007.

[10] J. Feng and A. K. Jain, “Fingerprint reconstruction: From minutiae to phase,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 2, pp. 209–223, Feb. 2011.