International Journal of Soft Computing 2 (2): 331-334, 2007

Data Integrity in Fingerprint Images

[1]A. Merry Ida, [2]V. Kavitha and [3]K.S. Easwarakumar
[1]M.E. Student, Noorul Islam College of Engineering, Kumaracoil, Tamilnadu, India
[2]Department of IT, Noorul Islam College of Engineering, Kumaracoil, Tamilnadu, India
[3]Department of CSE, Anna University, Chennai, Tamilnadu, India

Abstract: In e-commerce applications Authentication is provided with the combination of user id and password that results in fraud and misuse. Automated biometrics technology in general and fingerprints in particular, provide an accurate and reliable authentication method. However, fingerprint based authentication requires accessing fingerprint images scanned remotely at the user's workstation, a potentially weak point in the security system. Even though the communication channel is encrypted, stored or synthetic fingerprint images might be fraudulently transmitted. To enhance security, embedding additional information directly in compressed fingerprint images can use data hiding technique. Integrity is incorporated while transmitting fingerprint for authentication. Data Integrity is achieved by hashing the features of fingerprint to that of Unique Person Identification (UPID).

Key words: Fingerprint authentication, data integrity, WSQ compression and data hiding

INTRODUCTION

Authentication is a fundamental component of human interaction with computers and it is crucial to network security. With the increase growth of Internet there is a need to restrict access to sensitive data to authorized persons. Reliable and accurate identification of people is extremely important in a number of business transactions as well as privileged information.

Fingerprint is the symbol of illiteracy. The unique advantage of biometrics is that identifying a person based on physiological or behavioral characteristics. Fingerprint based authentication systems have been used for more than a century in law enforcement agencies. One of the obvious applications of biometrics is network security.

With the rapid growth of the Internet, transactions in electronic commerce amounts several billion US dollars. At present, service providers using combinations of user-id and password authenticate the buyers. The critical information about the transaction, such as the credit card number and the amount are send over the web using secure encryption method. So the credit card owners and credit card issuers will demand more reliable and secure authentication techniques.

Automated biometrics can help here. A typical fingerprint image is in the order of 512*512 pixels with 256 gray levels, resulting in an image size of 256 Kbytes. Both in web based or other on line transaction processing systems, it is undesirable to send uncompressed fingerprint images to the server. Thus standard image compression methods may be used. But these methodshave a tendency to distort the high frequency spatial structural ridge feature of a fingerprint image (Ahmad *et al*., 1999). This has lead to several research proposals regarding domain specific compression methods.

FBI has proposed an open Wavelet based image compression scheme (WSQ, 1993). This has become the de facto standard in the industry because of its low distortion even at very high compression ratios, but as WSQ is an open compression standard, transmitting a WSQ compressed image over the Internet is not secure. This is because if a compressed fingerprint image bit stream can be freely intercepted, it can be decompressed using readily available software and the purpose of using a biometric for encryption will become useless.

To enhance security, embedding additional information directly in compressed fingerprint images can use data hiding technique (Bendr *et al*., 1996). This method hides messages with minimal impact on the decompressed appearance of the image. Also, the image is not hidden in a fixed location, but is deposited in different places based on the structure of the image. Data Integrity is achieved by hashing the features of fingerprint to that of Unique Person Identification (UPID).

Corresponding Author: A. Merry Ida, M.E. Student, Noorul Islam College of Engineering, Kumaracoil, Tamilnadu, India

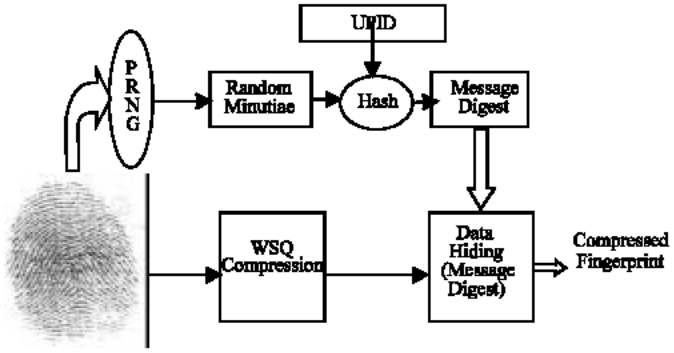

Fig.. 1: Architecture for authentication and integrity Encoder side

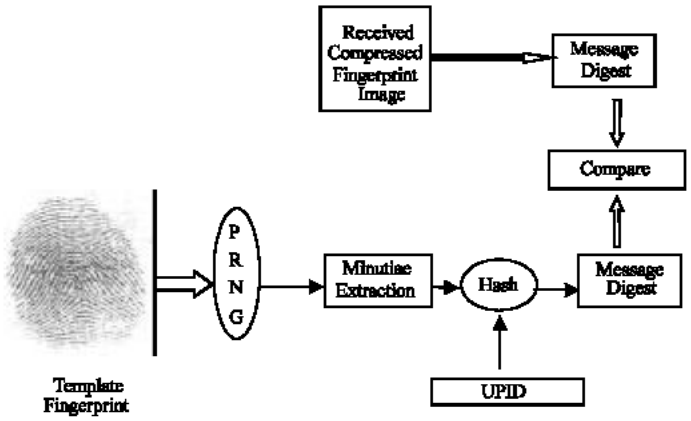

Fig. 2: Architecture for authentication and integriv Decoder

Data integrity: Identification of fingerprint is done by minutiae extraction. There are about hundred of minutiae and thirteen to fifteen minutiae are enough for identifying a person. Data Integrity is applied by choosing minutiae using pseudo random number generator. The chosen thirteen to fifteen minutiae and Unique Person Identification (UPID) are hashed to form the message digest. Message digest can be developed by other algorithms like SHA-1, etc. On the other hand the fingerprint is compressed using Wavelet Scalar Quantization Algorithm WSQ. The message digest is hidden in the compressed image using Data Hiding Techniques. The architecture for authentication and Integrity in the encoder side is depicted in Fig. 1.

In the receiver side thirteen to fifteen minutiae is extracted from the template fingerprint using Pesudo random Number generator and are hashed with the User Personal Identification Number to from the Message Digest. The Message Digest retrieved from the transmitted fingerprint image is compared for authentication and Data integrity, if matched the message conveyed is accepted, otherwise the message is rejected. The pseudo random number generator is installed by the corresponding authority for selecting minutiae in the client side and is assumed that the fingerprint-capturing device is secure. Figure 2 shows the architecture for authentication and integrity in decoder side. Various Steps involved are explained as below.

WSQ fingerprint compression: WSQ is an image compression technique whose goal is to reduce the number of bytes of a fingerprint image, sacrificing its quality as little as possible. It was designed targeting fingerprint images specifically, using their spectral characteristics to optimize the compression process. The process is lossy, meaning that the reconstructed image isn't equal to the original; some information being lost. The WSQ algorithm was specially designed to minimize the loss of information, so that the reconstructed image is as close as possible to the original; the differences

frequently being imperceptible. WSQ compression technology allows the compression of fingerprint images up to fifteen times (15:1) while maintaining their integrity for a visual comparison by a trained fingerprint examiner.

The input image is first decompressed in to 64 spatial frequency sub bands using perfect reconstruction multi-rate filter banks. The filters are implemented as a pair of separable 1D filter. The sub bands are the filter outputs obtained after a desired level of cascading filters. An interesting aspect of WSQ algorithm is the way it handles the image at the boundary. Instead of simply periodizing the image at the boundaries in both dimensions the standard specifies Symmetric Extension Transforms (SET) which essentially mirrors image across the boundaries. By extrapolating the signal in this way, the discrete wavelet transforms results in the same number of coefficients as the image size.

The second stage of WSQ encoding is the quantization process. The process of rounding high precision numbers into lower precision numbers with fewer digits is called quantization. Here the Discrete Wavelet Transforms (DWT) coefficients are transformed to integers with a small number of discrete values. This is accomplished by uniform scalar quantization for each sub band. These are two characteristics for each bandwidth of the bins (Q_k) and zero of the band (Z_k). Both Z_k and Q_k for each band are transmitted to decoder.

The final stage in WSQ encoding is Huffman coding of integer indices. For this purpose the bands are grouped into three blocks. In each block, the integer coefficients are re-mapped to numbers between 0-255 as per a translation table described in the standard.

This transaction table encodes run lengths of zeros and large values. The negative numbers are also translated in a similar way. The data-hiding algorithm works in the indices obtained after translation. It is assumed that the message size is very small compared to image size. The Huffman coding characteristics and table are not changed; the table can be computed as for the original coefficients.

Minutiae extraction: Normally fingerprint contains nearly hundred minutiae. From those 100 min 13 to 15 min can be used to identify a person. The proposed idea is to extract the minutiae from the fingerprint using the minutiae extraction algorithm. Steps involved in extracting minutiae from the fingerprint images are

Fingerprint binarization: Binarization the process of converting the gray-scale image into a binary image is called binarization. Zeros and ones forms binary image. Global Threshold Algorithm is used for performing binarization process. Looking at each pixel on the fingerprint image and deciding whether it should be converted to black (0) or white (255) i.e., to 0 or 1 from grey level to black and white image each pixel value is compared with a threshold value to make decision. If the pixel value is less than the threshold level then pixel value is set to Zero; otherwise it is set to 255.

Thinning: Consecutive fast parallel thinning algorithms are applied, in order to reduce the width of the ridges to a single pixel. These operations are necessary to simply the subsequent structural analysis of the image for the extraction of the fingerprint minutiae. Thinning is performed without modifying the original structure of the image. It should be taken care that miscalculations in beginnings, endings and/or bifurcation of the ridges, neither ridge can be broken. The parallel thinning algorithm reduces the thickness of ridges of the fingerprint image to unity.

Minutiae extraction: A feature extractor finds the ridge endings and ridge bifurcations from the input fingerprint images. If ridges can be perfectly located, then minutiae extraction is just a trivial task of extracting singular points in a tinned ridge map. However in practice, it is not always possible to obtain a perfect ridge map. The minutiae are extracted by Crossing Number (CN) algorithm.

After extracting the minutiae, pseudo random generator algorithm is used to select 13 to 15 min randomly. Hash function is used to create message digest. The message digest created is of fixed size. The selected minutiae and the user ID given by the user at the time of processing are XORed (Hash Function) to form the message digests, which is of fixed length. Then the message digest is embedded in the fingerprint using data-hiding algorithm.

Data hiding: Let M be the message digests in bits. M is assumed to be very small compared to the number of pixels in the image. The data-hiding algorithm follows three stages, Site Selection, Random Number Seeding, Message Setting.

Site selection set S: Given the partially converted quantized integer indices, the role of this stage is to collect the indices of all possible sites where a change in the least significant bit is tolerable. The translated indices of the quantizer have special codes to reflect run lengths of zeros and large integer values and other

control sequences. These sites are avoided. In this implementation, the translated indices ranges from 107 to 254 are used, but 180 are excluded.

Random number seeding: The sites from set S are selected and modified in a random fashion. To retain predictability in coder and decoder, it is decided to select the seed for the random number generator from the sub bands that do not change. For example, in the selection process, the contents of sub bands 0-6 are left unchanged to minimize distortion. In principle, any statistic can be chosen from their bands including the values at fixed places in these bands. This seed selection ensures that both the message is embedded in random locations and that the message can be read only if seed selection algorithm is known.

Message setting: Each bit of message digest is stuffed into the sites chosen randomly from the list of sites. That is, a random site from set S is chosen as per the pseudo random generator and its low order bits are changes to be identical to the current message bit.

$$\text{For } i = 1,2,\ldots,m,$$

Let b_i be the i^{th} message bit,

Choose a random site as per the pseudo-random generator; let
the index value be Ind_k

$$Ind_k \text{ bit } 0 = m_i$$

Message bit I is the least significant bit of the value at set k. In spite of embedding information regarding data integrity, the above methodology allows perfect reconstruction of the image. The image after decompression and extraction of the message digest is nearly the same as the template fingerprint image. The error due to the embedded message is not perceptually significant and does not affect subsequent processing and authentication.

Fig. 3: (a) Input fingerprint image (b) Embedded message Digest fingerprint

RESULTS

Input fingerprint image is shown in Fig. 3a. After reconstruction of the fingerprint image in decoder side is shown in Fig. 3b. The reconstructed image is same as the original image. We can hide a message digest of considerable length in the image, probably sufficient for e-commerce transactions.

The message digest is hidden by using robust data hiding algorithm in the wavelet compressed finger print images. This model can be easily extended to other compressed image such as medical images and satellite images.

REFERENCES

Ahmed, S. Abutaleb M. and Kamel, 1999. A genetic algorithm for the estimation of ridges in fingerprints, IEEE. Trans. Image Processing, 8: 8.

Bender, W.D., Gruhl, N. Morimoto and A. Lu, 1996. Techniques for Data Hiding, IBM. Sys. J., 35: 3-4.

WSQ, 1993. Gray-Scale fingerprint image compression specification.