Hiding Fingerprint Minutiae in Images

Anil K. Jain and Umut Uludag
Computer Science and Engineering Department, Michigan State University
3115 Engineering Building, East Lansing, MI, 48824, USA
{jain, uludagum}@cse.msu.edu

Abstract

We introduce an application of steganography and watermarking to enable secure biometric data (e.g., fingerprints) exchange. We hide fingerprint minutiae data in a host image, which can be a synthetic fingerprint image, a face image or an arbitrary image. It is this carrier image that is transferred to the receiving party in this exchange, instead of the actual minutiae data. The hidden biometric data are extracted accurately from the carrier image using a secret key. Furthermore, when the host is a face image, the proposed method provides an additional cue in authenticating the user. Data are hidden in the host image in an adaptive way to minimize possible degradations to that image. Our method can also tolerate several attacks on the carrier image.

1. Introduction

Biometrics-based personal identification techniques, which use physiological or behavioral characteristics, are becoming increasingly popular compared to traditional token-based or knowledge-based techniques such as identification card (ID), passwords, etc. One of the main reasons for this popularity is the ability of the biometrics technology to differentiate between an authorized person and an impostor who fraudulently acquires the access privilege of an authorized person [4].

Among various commercially available biometric techniques such as face, facial thermogram, fingerprint, iris, etc., fingerprint-based techniques are the most extensively studied and the most frequently deployed. A fingerprint-based biometric system has four stages: acquisition, representation, feature extraction and matching. In the acquisition stage, a fingerprint image is captured via inked or live-scan methods. In most of the recent civilian applications, live-scan methods that directly produce the digital image of fingerprints are used. In the representation stage, the aim is to find invariant and discriminatory information inherent in the fingerprint image. In minutiae-based systems, the discontinuities in the regular ridge structure of fingerprint images, called ridge endings and ridge bifurcations, are identified in feature extraction stage. During matching, a similarity value between the features extracted from the template and the input fingerprint images is calculated. This similarity value is used to arrive at an accept/reject decision [4], [5].

While biometrics techniques have inherent advantages over traditional personal identification techniques, the problem of ensuring the security and integrity of the biometrics data is critical. For example, if a person's biometric data (e.g., his/her fingerprint image) is stolen, it is not possible to replace it unlike replacing a stolen credit card, ID or password. Schneier [12] points out that a biometrics-based verification system works properly only if the verifier system can guarantee that the biometric data came from the legitimate person at the time of enrollment. Furthermore, while biometrics data provide uniqueness, they do not provide secrecy. For example, a person leaves fingerprints on every surface he/she touches and face images can be surreptitiously observed anywhere that person looks. Ratha *et al.* [9] identify eight basic sources of attacks that are possible in a generic biometric system. Examples of these attacks include presenting a “fake” finger at the sensor and intercepting the transmitted template and/or sensed fingerprint. All of these attacks have the possibility to decrease the credibility of a biometric system.

In order to promote the wide spread utilization of biometric techniques, an increased security of the biometric data, especially fingerprints, seems to be necessary. Encryption, watermarking and steganography are possible techniques to achieve this. Steganography, derived from the Greek language and meaning secret communication, involves hiding critical information in unsuspected carrier data. While cryptography focuses on methods to make encrypted information meaningless to unauthorized parties, steganography is based on concealing the information itself. Digital watermarking techniques can be used to embed proprietary information, such as company logo, in the host data to protect the intellectual property rights of that data [3], [13]. They are also used for multimedia data authentication. Encryption does not provide security once the data is decrypted. On the other hand, since watermarking involves embedding information into the host data itself, it can provide security even after decryption. Furthermore, encryption can be applied to the watermarked data. Another option is to use steganograpy: by hiding fingerprint features in a carrier image, the security of fingerprint information can be increased.

In this paper, two applications of an amplitude modulation-based watermarking method are introduced. Section 2 contains an introduction to watermarking techniques along with a brief summary of current research on fingerprint image watermarking. Section 3 outlines the proposed applications and includes the details of the watermarking method. In Section 4, we present experimental results on several host images. The robustness of the method to several attacks is also analyzed. Conclusions and future research directions are presented in Section 5.

2. Watermarking Techniques

Digital watermarking, or simply watermarking, which is defined as embedding information such as origin, destination, access level, etc. of multimedia data (e.g., image, video, audio, text, etc.) in the host data, has been a very active research area in recent years [3], [13]. General image watermarking methods can be divided into two groups according to the domain of application of watermarking. In spatial domain methods (e.g., [6]), the pixel values in the image channel(s) are changed. In spectral-transform domain methods, watermark signal is added to the host image in a transform domain such as the full-frame DCT domain [1], Fourier-Mellin domain [11], etc.

There have been only a few published papers on watermarking of fingerprint images. Ratha *et al.* [10] proposed a data hiding method, which is applicable to fingerprint images compressed with WSQ wavelet-based scheme. The discrete wavelet transform coefficients are changed during WSQ encoding, by taking into consideration possible image degradation. Message bits are encoded as the least significant bits of selected coefficients. Pankanti and Yeung [7] proposed a fragile watermarking method for fingerprint image verification. A spatial watermark image is embedded in the spatial domain of a fingerprint image by utilizing a verification key. The proposed method can localize any region of image that has been tampered. To increase the security of the watermark data, the original watermark image is first transformed into another mixed image, and this mixed image is used as a new watermark image. The mixed image does not have a meaningful appearance, contrary to original watermark image that can contain specific logos or texts. Pankanti and Yeung conclude that their watermarking technique does not lead to a significant performance loss in fingerprint verification.

Uludag *et al.* [15] described two spatial domain watermarking methods for fingerprint images. The first method utilizes gradient orientation analysis in watermark embedding; pixel values at watermark embedding locations are changed in a way to preserve the quantized gradient orientations around those pixels. As a result, the watermarking process alters none of the features extracted using gradient information. The second method preserves the singular points in the fingerprint image, so the classification of the watermarked fingerprint image (e.g., into arch, left loop, etc.) is not affected.

3. Hiding Minutiae Data

In this paper, we consider two application scenarios. The basic data hiding method is the same in both of the scenarios, but it differs in the characteristics of the host image carrying the minutiae data and the medium of data transfer.

3.1. Application Scenarios

The first scenario involves a steganography-based application (Figure 1): the data (fingerprint minutiae) that need to be transmitted (possibly via a non-secure communication channel) is hidden in a host (i.e., cover) image, whose only function is to carry the data. In the following, host, carrier and cover image terms will be used interchangeably. The host image is not related to the hidden data in any way. As a result, the host image can be any image available to the encoder. In our application, we consider three different types of cover image for this purpose: synthetic fingerprint images, face images and arbitrary images. Figure 2 shows examples of these cover images. The synthetic fingerprint image is obtained after a post-processing of the image generated using the algorithm described by Cappelli *et al.* [2]. The post-processing included increasing the image size from 320x240 to 360x280, eliminating the dominance of white background, and replacing this background with a uniform gray level distribution between [225, 235], with a mean of 230. This background transformation helps in hiding the minutiae data more invisibly. Using such a synthetic fingerprint image to carry actual fingerprint minutiae data provides an increased level of security since a person who intercepts the communication channel and obtains the carrier image would treat this synthetic image as a real fingerprint image! The face image (384x256) was captured in our laboratory. The "Sailboat" image (512x512) is from USC-SIPI database [14]. The luminance channel of this color image is used as the cover image. This application can be used to counter the attack

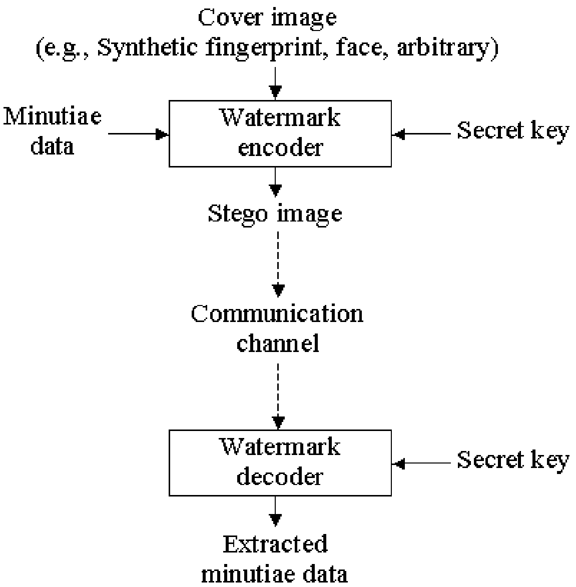
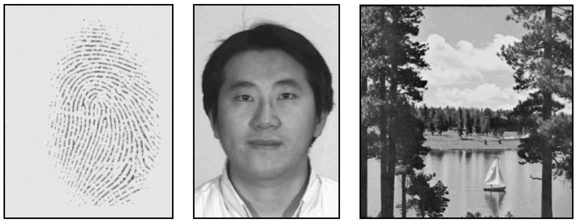


Figure 1. Steganograpy-based minutiae hiding.

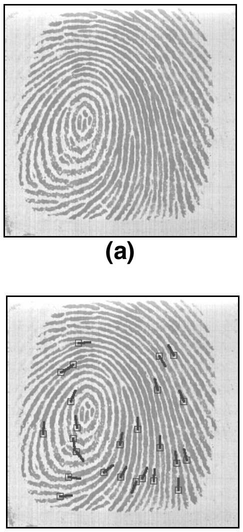


(a) (b) (c)

Figure 2. Sample cover images: (a) synthetic fingerprint, (b) face, (c) “Sailboat”.

on the communication channel between the database and the fingerprint matcher. An attacker will most probably not suspect that a cover image is carrying the minutiae information.

Figure 3 shows an input fingerprint image, overlaid minutiae image and the attributes (x, y, θ) of the extracted minutiae. These attributes constitute the data to be hidden in the host images. The fingerprint images (300x300) used here were captured by a solid state sensor manufactured by Veridicom. The minutiae are extracted using the method outlined in [5]. The minutiae data shown in Figure 3(c) contain three fields per minutiae: x-coordinate, y-coordinate and orientation, for a total of 25 minutiae. The minutiae data $((x_i, y_i, \theta_i),\ i = 1, 2, ..., 25)$ are hidden in the cover image using the method explained in Section 3.2. A secret key is utilized in encoding to increase the security of the hidden data. The image with embedded data (stego image) is sent through the channel that may be subject to interceptions. At the decoding site, using the same key that was used by the encoder (which can be delivered to the decoder using a secure channel prior to stego image transfer), the hidden data is recovered from the stego image.



(a)

(b)

x	y	θ
76	216	242
121	195	255
136	82	292
136	229	248
170	90	262
172	169	270
178	46	274
184	85	82
192	146	281
196	198	270
201	89	52
212	233	255
216	220	262
228	125	321
234	79	8
234	147	298
236	175	295
240	167	112
259	68	356
60	92	356
77	197	58
88	85	144
98	69	332
239	190	274
251	222	270

(c)

Figure 3. Minutiae data: (a) input fingerprint image, (b) overlaid minutiae image, (c) minutiae point attributes.

The second scenario aims at increasing the security of face images. In this scenario, a person’s face image, which also carries that person’s fingerprint minutiae data, is encoded in a smart card (Figure 4). At a controlled access site, this image will be read from the smart card and the original face image will be reconstructed. The extracted minutiae data will be compared to the minutiae obtained from the user at the access site. These two minutiae data sets and the reconstructed face image will be used to accept or reject the user. This application can be used to eliminate several types of biometric system attacks described in [9]. Fake biometric submission via a smart card that contains a non-authentic image will be useless since that image will not contain the true minutiae data. Resubmission of digitally stored biometrics data (e.g., via a stolen but authentic smart card) will not be feasible since the system authenticates every user by using this data along with the minutiae data obtained online at a controlled access site. A user who succeeds in inserting a new template into the database will not be authenticated at the access site since this new template will not contain the minutiae data.

3.2. Data Hiding Method

The amplitude modulation-based watermarking method described here is an extension of the blue channel watermarking method of Kutter *et al.* [6]. The proposed method includes image adaptivity and watermark strength controller along with the basic method in [6]. An earlier version of the method is presented in [16], in which the increase in data decoding accuracy related to these

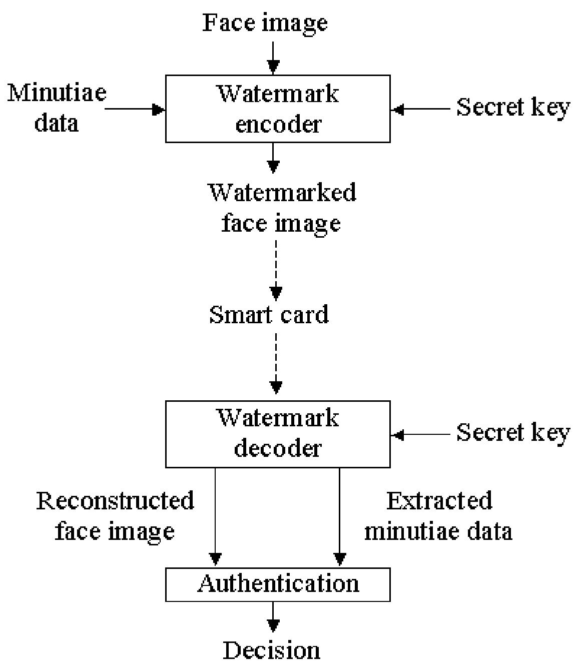


Figure 4. Minutiae hiding for face verification.

extensions is analyzed. In the first step, the minutiae data to be hidden is converted into a bit stream. Every field of individual minutia is converted to a 9-bit binary representation. Such a representation can code integers between [0, 511] and this range is adequate for x-coordinate ([0, #rows-1]), y-coordinate ([0, #columns-1]) and orientation ([0, 359]) of a minutia. A random number generator initialized with the secret key generates locations of the host image pixels to be watermarked. These pixels are changed according to the following equation

$$P_{WM}(i,j) = P(i,j) + (2s-1)P_{AV}(i,j)q\left(1+\frac{P_{SD}(i,j)}{A}\right)* \left(1+\frac{P_{GM}(i,j)}{B}\right), \quad (1)$$

where $P_{WM}(i,j)$ and $P(i,j)$ are values of the watermarked and original pixels at location (i,j), respectively. The value of watermark bit is denoted as s and watermark embedding strength is denoted as q, $s \in [0,1]$, $q > 0$. $P_{AV}(i,j)$ and $P_{SD}(i,j)$ denote the average and standard deviation of pixel values in a neighborhood of location (i,j) and $P_{GM}(i,j)$ denotes the gradient magnitude at (i,j). A and B are weights for the standard deviation and gradient magnitude, respectively. In our experiments, P_{AV} is calculated in a 5x5 square neighborhood and P_{SD} is calculated in a 5x5 cross-shaped neighborhood. The gradient magnitude is computed via the 3x3 Sobel operator.

These image adaptivity terms adjust the magnitude of watermarking, by utilizing several properties of the human visual system (HVS). Using $P_{AV}(i,j)$ in modulating watermark magnitude conforms to amplitude nonlinearity of HVS. As Eq. (1) shows, the magnitude of the change in the value of pixel (i,j) caused by watermarking is higher when the $P_{AV}(i,j)$ value is high. Standard deviation and gradient magnitude terms utilize contrast/texture masking properties of HVS. Masking is the reduction of the visibility of an image component (masked signal) due to the presence of another component [8]. The changes in pixel values in highly textured and high contrast image areas are masked more strongly than changes in smooth image areas. These image adaptivity terms increase the magnitude of watermarking in image areas where such an increase does not become very visible to a human observer. This leads to more accurate decoding of the hidden data, especially in the case of attacks on host images.

Every watermark bit with value s in Eq. (1) is embedded at multiple locations (such as 30 locations/bit) in the host image. This redundancy increases the correct decoding rate of the embedded information. The amount of this redundancy is limited by image capacity (size) and visibility of the changes in pixel values.

In addition to the binary minutiae data, two reference bits, 0 and 1, are also embedded in the image. These reference bits help in calculating an adaptive threshold in determining the minutiae bit values during decoding. Decoding starts with finding the data embedding locations in the watermarked image, via the secret key used during the watermark encoding stage. For every bit embedding location, (i,j), its value is estimated as the linear combination of pixel values in a 5x5 cross-shaped neighborhood of the watermarked pixels as in Eq. (2).

$$\hat{P}(i,j) = \frac{1}{8}\left(\sum_{k=-2}^{2} P_{WM}(i+k,j) + \sum_{k=-2}^{2} P_{WM}(i,j+k) - 2P_{WM}(i,j)\right), \quad (2)$$

The difference between the estimated and watermarked pixel values is calculated by Eq. (3) as

$$\delta = P_{WM}(i,j) - \hat{P}(i,j), \quad (3)$$

These differences are averaged over all the embedding locations associated with the same bit, to yield $\overline{\delta}$. For finding an adaptive threshold, these averages are calculated separately for the reference bits, 0 and 1, as $\overline{\delta}_{R0}$ and $\overline{\delta}_{R1}$, respectively. Finally, the watermark bit value $\hat{s}$ is estimated as

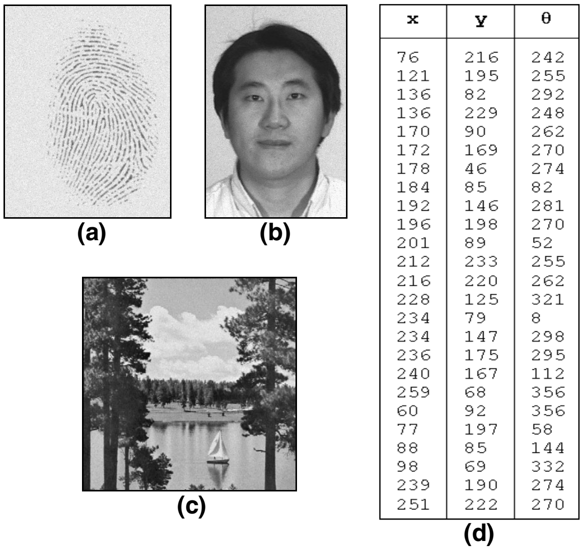
$$\hat{s} = \begin{cases} 1 & \text{if } \ \bar{\delta} > \dfrac{\bar{\delta}_{R0} + \bar{\delta}_{R1}}{2} \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

The watermark decoding process can produce erroneous bits since decoding is based on an estimation procedure, which may fail to find the exact original pixel values. In order to increase the decoding accuracy, the encoder uses a controller block. This block adjusts the strength of watermarking, q, on a pixel-by-pixel basis, if there is a possibility of incorrect bit decoding. From decoded watermark bits, the minutiae data hidden in the host image is extracted. In our second application scenario, which typically includes a smart card containing a digital face image marked with the minutiae data of the person depicted in this image, an estimate of the original face image is also found via replacing the watermarked pixel values with the $\hat{P}(i,j)$ estimate calculated by Eq. (2).

4. Experimental Results

Figures 5 (a)-(c) show the stego images, which carry the minutiae data shown in Figure 3(c), for the cover images in Figure 2. Nearly 17% of the stego image pixels are changed during data hiding, for all the three cover images. The key used in generating the locations of the pixels to be watermarked is selected as the integer 1,000. However, the exact value of key does not affect the performance of the method. Other watermarking parameters are set to: $q = 0.1$, $A = 100$, $B = 1000$. A higher q value increases the visibility of the hidden data. Increasing A or B decreases the effect of standard deviation and gradient magnitude in modulating watermark embedding strength, respectively. The hidden data size is approximately 85 bytes. The extracted minutiae data from all the three cover images is shown in Figure 5(d); it is exactly the same as the hidden data shown in Figure 3(c).

For the second application scenario, the face image shown in Figure 6(a) is watermarked using the same minutiae data (Figure 3(c)). Other parameters of the watermarking algorithm are the same as the ones used for the first application scenario. The extracted minutiae data are identical to the embedded data. Figure 6(c) shows the negative image of the difference between original image and the watermarked image. Furthermore, the reconstructed face image is given in Figure 6(d).



x	y	θ
76	216	242
121	195	255
136	82	292
136	229	248
170	90	262
172	169	270
178	46	274
184	85	82
192	146	281
196	198	270
201	89	52
212	233	255
216	220	262
228	125	321
234	79	8
234	147	298
236	175	295
240	167	112
259	68	356
60	92	356
77	197	58
88	85	144
98	69	332
239	190	274
251	222	270

Figure 5. Stego images and decoded data: (a) synthetic fingerprint, (b) face, (c) "Sailboat", (d) extracted minutiae data.

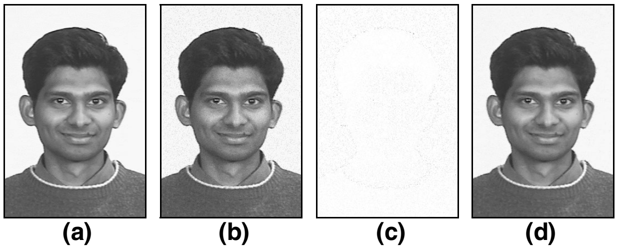


Figure 6. Watermarking for user verification: (a) input face image, (b) watermarked face image, (c) negative of difference image, (d) reconstructed face image.

In order to assess the average magnitude of changes in pixel values, several image and watermarking characteristics for the four host images are computed (see Table 1). The first column is the average value for all the image pixels. The second column is the average value for the changed (i.e., watermarked) pixels. The last column shows the average change in values (due to watermarking) of the watermarked pixels.

Table 1. Host image and watermarking characteristics.

Host Image	Overall Pixel Average	Changed Pixel Average	Avg. Change in Pixel Values
Figure 2(a)	221.8	222	24.9
Figure 2(b)	150.2	150.6	15.6
Figure 2(c)	124.8	124.9	15.4
Figure 6(a)	159.6	160.2	14.5

Additional tests to determine the performance of the proposed algorithm were conducted as follows: 15 images (5 synthetic fingerprint, 5 face, 5 arbitrary) were watermarked with 5 different minutiae data, by using 5 different keys. As a result, 375 different watermarked images are produced. Characteristics and sources of the host images and watermarking parameters are the same as given previously. Individual minutiae data sets contained between 23 to 28 points, with an average of 25 points. From all of these 375 watermarked images, we were able to extract the embedded minutiae information with 100% accuracy.

The proposed data hiding method is robust and can tolerate certain types of attacks, namely image cropping and JPEG compression. The hidden minutiae data are extracted correctly from (i) 40% cropped and (ii) JPEG compressed (quality factor 90) versions of all the four watermarked images (Figures 5 (a)–(c) and Figure 6(b)). Figure 7 shows the attacked images for three of these host images.

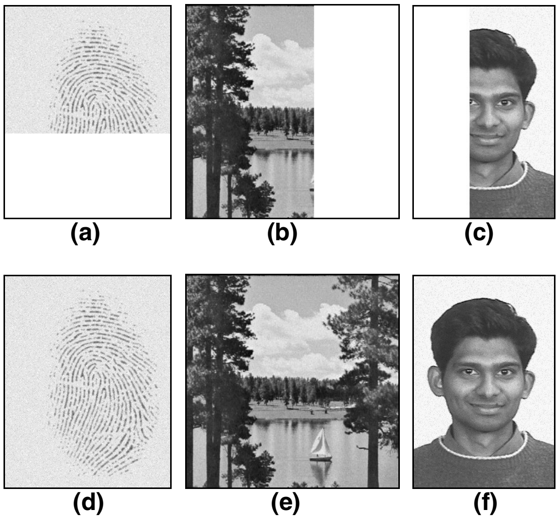


Figure 7. Robustness to attacks: (a), (b) and (c) are 40% cropped versions of watermarked images, (d), (e) and (f) are JPEG compressed watermarked images.

5. Conclusions

The ability of biometrics-based personal identification techniques to differentiate between an authorized person and an impostor who fraudulently acquires the access privilege of an authorized person is one of the main reasons for their popularity compared to traditional identification techniques. However, the security and integrity of the biometric data itself is an important issue. Encryption, watermarking and steganography are possible techniques to secure biometrics data. In this paper, two applications of a watermarking method are presented. The first application is related to increasing the security of biometric data exchange, which is based on steganography. In the second application we authenticate a user based on his face image, along with the fingerprint information hidden in the face image. The proposed method utilizes several properties of the human visual system to keep the visibility of the changes made to the host image low. The data decoding performance in the case of several attacks on host images is also analyzed. Currently, we are working on analyzing the data hiding capacity of host images. Furthermore, we are expanding the proposed method to hide data in real fingerprint images.

References

[1] M. Barni, F. Bartolini, V. Cappellini and A. Piva, "A DCT domain system for robust image watermarking", *Signal Processing*, vol. 66, no. 3, May 1998, pp. 357-372.

[2] R. Cappelli, A. Erol, D. Maio and D. Maltoni, "Synthetic fingerprint image generation", *Proc. ICPR*, Sept. 3-7, 2000, Barcelona, vol. 3, pp. 475-478.

[3] F. Hartung and M. Kutter, "Multimedia watermarking techniques", *Proc. IEEE*, vol. 87, no. 7, July 1999, pp. 1079-1107.

[4] A.K. Jain, L. Hong and S. Pankanti, "Biometric identification", *Comm. ACM*, vol. 43, no. 2, Feb. 2000, pp. 91-98.

[5] A.K. Jain, L. Hong, S. Pankanti and R. Bolle, "An identity-authentication system using fingerprints", *Proc. IEEE*, vol. 85, no. 9, Sept. 1997, pp. 1365-1388.

[6] M. Kutter, F. Jordan and F. Bossen, "Digital signature of color images using amplitude modulation", *Proc. SPIE EI*, San Jose, Feb. 1997, vol. 3022, pp. 518-526.

[7] S. Pankanti and M.M. Yeung, "Verification watermarks on fingerprint recognition and retrieval", *Proc. SPIE EI*, San Jose, Jan. 1999, vol. 3657, pp. 66-78.

[8] T.N. Pappas and R.J. Safranek, "Perceptual Criteria for Image Quality Evaluation", *in Handbook of Image and Video Processing*, Al Bovik (ed.), Academic Press, San Diego, 2000, pp. 669-684.

[9] N.K. Ratha, J.H. Connell and R.M. Bolle, "An analysis of minutiae matching strength", *Proc. 3rd AVBPA*, Halmstad, Sweden, June 2001, pp. 223-228.

[10] N.K. Ratha, J.H. Connell and R.M. Bolle, "Secure data hiding in wavelet compressed fingerprint images", *Proc. ACM Multimedia 2000,* pp. 127-130.

[11] J.J.K. Ruanaidh and T. Pun, "Rotation, scale and translation invariant spread spectrum digital image watermarking", *Signal Processing*, vol. 66, no. 3, May 1998, pp. 303-317.

[12] B. Schneier, "The uses and abuses of biometrics", *Comm. ACM*, vol. 42, no. 8, Aug. 1999, pp. 136.

[13] M.D. Swanson, M. Kobayashi and A.H. Tewfik, "Multimedia data-embedding and watermarking technologies", *Proc. IEEE*, vol. 86, no. 6, June 1998, pp. 1064-1087.

[14] The USC-SIPI Image Database. [Online]. http://sipi.usc.edu/services/database/Database.html.

[15] U. Uludag, B. Gunsel and M. Ballan, "A spatial method for watermarking of fingerprint images", *Proc. 1st Intl. Workshop on Pattern Recognition in Information Systems*, Setúbal, Portugal, July 2001, pp. 26-33.

[16] U. Uludag, B. Gunsel and A.M. Tekalp, "Robust watermarking of busy images," *Proc. SPIE EI*, San Jose, Jan. 2001, vol. 4314, pp. 18-25.