Vulnerabilities in biometric encryption systems

Andy Adler
School of Information Technology and Engineering,
University of Ottawa
Ottawa, Ontario, Canada

adler@site.uOttawa.ca

ABSTRACT

Biometric encryption systems embed a secret code within a biometric image in a way that it can be decrypted with an image from the enrolled individual. We describe a potential vulnerability in biometric encryption systems that allows a less than brute force regeneration of both the secret code and an estimate of the enrolled image. This vulnerability requires the biometric comparison to "leak" some information from which an analogue for a match score may be calculated. Using this match score value, a "hill-climbing" attack is performed against the algorithm to calculate an estimate of the enrolled image, which is then used to decrypt the code. Results are shown against a simplified implementation of the algorithm of Soutar et al. (1998). Possible extensions of this attack to other biometric encryption algorithms are discussed.

1.0 INTRODUCTION

There have been significant recent advancements in algorithms for biometric encryption (Uludag et al, 2004). Biometric encryption systems embed a secret code into the template, in such a way that it can be decrypted only with an image from the enrolled individual. In contrast, traditional biometric technology simply tests for a match between a new image presented and a stored biometric template - a compact digital representation of the essential biometric features. While biometric encryption systems are not widely deployed, they appear to offer some compelling benefits for many applications (Davida et al. 1998). Typically, biometric authentication systems are integrated into sophisticated security systems, in which the biometric subsystems form a relatively small part. Using a traditional biometric system, the input image from the user is compared to that encoded in an enrolled template, and if they are sufficiently similar, the biometric subsystem releases stored security tokens or codes to activate further processing. In such a system, an attack could allow access to both the biometric template and the security codes. This could potentially allow both access to the system (using the security tokens), and identification of the user (from the template). In contrast, a system using biometric encryption avoids this problem, since the secret code is bound to the biometric template in a way that an attacker should not be able to easily determine the biometric image or the codes by analysing the template.

The benefit of biometric encryption is perhaps most important for mobile applications of biometrics. For example, biometric identity cards, such as those designed into many new national passports, contain biometric templates. It is reasonable to assume that the template data will be accessible to someone with physical access to the documents, since the protocol for access to these data will, at a minimum, be available to governments and the manufacturers of passport readers. Another important application of biometric encryption is for control of access to digital content, with the primary interest being in preventing copyright infringement. Digital documents encoded with the biometric of the user(s) with approved access will presumably be subject to attacks, especially since both the documents and the software to access them will be widely distributed. Issues in the design of such digital content protection systems are discussed in a recent special issue of the Proceedings of the IEEE: "Special Issue on Enabling Security Technologies for Digital Rights Management" (Vol. 92, No. 6, June 2004).

The primary difficulty with biometric encryption systems is the variability in the biometric image between data measurements. For example, a fingerprint image changes with variations in the sweat, oil, and dirt on the skin; cuts and other damage; changes in body fat; and with many other factors. Such variability, of course, is the primary challenge in the design of biometrics algorithms. In the case of biometric encryption, it means that an image cannot be treated as a code by itself, since it varies with each presentation. For biometric encryption systems, this variability becomes especially difficult. An algorithm must be designed which allows one image, with certain significant differences from the original, to decode the complete secret code, while another image, only slightly more different form the original, must not allow decoding (or "leaking") of any information.

This paper considers one possible approach to attacking algorithms based on biometric encryption. If an algorithm does not completely prevent information leak from non-matching images, it may be possible to gather such information over many iterations. Thus, it may be possible to use the information in the encrypted template to estimate a sufficiently accurate image that can correctly match against the enrolled image. This rest of this paper presents some early analysis that suggests that such an attack is indeed possible.

2.0 IMAGE RECONSTRUCTION FROM TEMPLATES

The proposed attack on biometric encryption is to use "leaked" information from the template to iteratively estimate an image of the enrolled biometric, which is then used to unlock the secret code. The essential ingredient for this attack is the existence of some information that can allow calculation of an analogue of a match score. As noted before, it is extremely difficult to design a biometric encryption algorithm to give complete information for a "close" answer, but no information for a slightly less accurate one. For this attack, the only requirement for such a match score is that it increases with the similarity of the input image to the one encoded.

Several authors have shown that, given access to match score data, it is possible to reconstruct a good estimate of an unknown enrolled image (Soutar et al., 1999) from a fingerprint (Hill, 2001) or face recognition template (Adler, 2003). In each case, an algorithm is developed which allows small, but physiologically realistic, modifications to be made to an input image, and used to perform a "hill-climbing" attack. The image is presented to the algorithm and compared against an unknown target to obtain a match score. Then, iteratively, modifications are made to the input, and those that increase the match score are retained. Eventually, a best-match image is generated, which resembles the essential features of the unknown target, and is able to compare to it at high match score. In order to protect against this attack, the BioAPI (2001) specifies that match scores should be quantized. However, recently, we (Adler, 2004) have shown that the hill-climbing attack can be modified to overcome the effects of quantization (for reasonable levels of quantization, ie. where one quantization level corresponds to a 10% change in match confidence).

Our results show that the modified hill-climbing algorithm is required for attacks against biometric encryption algorithms. This appears to be because match scores calculated from biometric encryption algorithms are not easily related to traditional biometric match score values, and often it is only possible to calculate a quantized value. For example, with an error correcting code, the match score may be the number of bits that need correction, resulting in a heavily quantized score. The next section describes the hill climbing algorithm (based on Adler, 2004). This algorithm has been shown to work successfully for face recognition systems; however, the work of Hill (2001) and Uludag (2004b) suggests that it is extensible to fingerprint biometrics.

The *target* is defined as the person whose image is to be regenerated. A software application was implemented with the goal of regenerating a face image of the *target* (IM_{targ}). The application has access to a local database of face images, and has access to a biometric comparison engine, in which the target template is stored. The software begins with only the ability to obtain match scores (MS) of the target compared to an arbitrarily chosen image (IM). We represent this function as:

$MS_i = biometric\_compare(\ IM, IM_{\text{targ}}\ )$

Using these values, the hill-climbing algorithm functions as follows:

1. *Local database preparation:* A local database (LI) of frontal pose face images is obtained. Images are rotated, scaled, cropped, and histogram equalized such that all images have the same size (150×200 pixels), eye locations (horizontal, vertical pixel coordinates of the left and right eyes of 50,90 and 100,90), and pixel intensity distribution.
2. *Eigenface calculation:* Using the local image database, LI, a principle components analysis decomposition is used to calculate an set of eigenimages (or eigenfaces) (Turk and Pentland, 1991), using the method of Grother (2000). The image is then divided into four quadrants (figure 1, left). Quadrant eigenimages ($EF_{i,\text{quadrant}}$) are then defined to be equal to EF_i within the quadrant and zero elsewhere. The edge of each quadrant is then smoothed to provide a gradual transition over 10 percent of the image width and height.
3. *Initial image selection:* In initial estimate (IM_0) is chosen which is subsequently iteratively improved in the next step. A selection of images is chosen randomly from the local database, LI, and individually compared to the target. IM_0 is selected to be the one with the highest match score.
4. *Iterative estimate improvement:* Iterate the following steps for step number i.
   A. Randomly select an eigenimage, EF_k
   B. Randomly select a quadrant Q. The diametrically opposite quadrant is referred to as OQ.
   C. Generate an image RN consisting of random Gaussian noise in OQ and zero elsewhere.
   D. We then calculate the contribution of RN which reduces the quantized match score by one quantization level.
      - Calculate $MS_i = biometric\_compare(\ IM_k, IM_{\text{targ}}\ )$
      - Using a bisection search, calculate a minimum value n and an associated noisy image $NI = IM_k + n \times RN$, such that the corresponding match score $MS_{NI} = biometric\_compare(\ NI, IM_{\text{targ}}\ )$ is less than MS_i
   E. Interate for j for a small range of values c_j, using the quadrant Q eigenimage.
      $MS_j = biometric\_compare(\ NI + c_j \times EF_{k,Q}, IM_{\text{targ}}\ )$
   F. Select $j_{\max}$ as the value of j for which MS_j is maximized.
   G. Calculate $IM_{i+1} = IM_i + c_{j\max} \times EF_{k,Q}$
   H. Truncate values to image limits (ie. 0 to 255) if any pixel values of IM_{i+1} exceed limits.

   Repeat iterations until there is no significant improvement in match score.

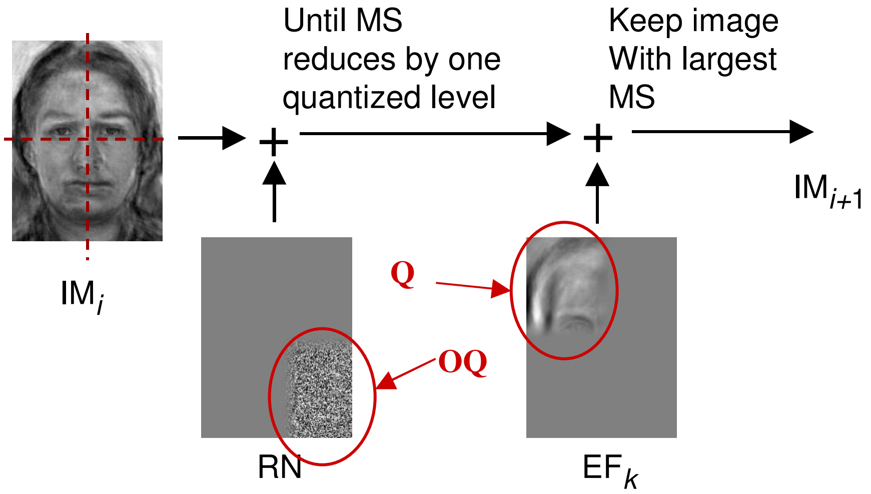

Figure 1: Schematic diagram of hill-climbing algorithm for quantized match scores. At each iteration, the candidate image is first "worsened" with the addition of random noise to a quadrant, until the match score is below a quantized level. Then a component of an eigenimage is added to the opposite quadrant, and the maximum match score output is retained.

This algorithm works separately on quadrants of the image. Because the quantized match score will not normally give information to allow hill climbing, a carefully chosen level of noise is introduced into the opposite image quadrant, in order to force the quantized score into a range where its information can once again be used. The local database does not need to resemble the target image, and may be one of the many freely available face image databases (for example Craw et al. (1999) and Phillips et al. (2000)).

3.0 BIOMETRIC ENCRYPTION SYSTEMS

This paper considers the fingerprint biometric encryption algorithm of Soutar et al. (1998). This algorithm was chosen because it represents concrete system which has been implemented and for which the details are well described. Bioscypt (the employer of Soutar) has indicated that significant enhancements were made to this algorithm after the published version. However, this paper simply presents a framework for an attack, and not necessarily a break of a specific, implemented, algorithm. For a review of other recent biometric encryption systems, refer to Uludag et al. (2004).

The enrolment process for this system takes several sample images, and a secret code, and creates a template binding the code to the images. This differs for some other systems, such as that of Davida et al. (1998), in which the biometric image forms a unique key. The system of Soutar et al. (1998) calculates a template related to the input image by frequency domain correlations. We describe a simplified operation of this system, using slight variations in notation from the original. During enrolment, an average image f_0 is obtained (with 2D Fourier transform $F_0(u)$) from multiple samples of input fingerprint, after suitable alignment. In order to encode the secret, a random code is chosen and encoded as a phase-only function $R_0(u)$ such that the amplitude is one and the phase is $\pm\pi/2$. Using F and R, a filter function $H(u)$ is calculated based on a Wiener inverse filter, as

$$H_0 = F^{*}R_0^{*} \,/\, (F^{*}F + N^2)$$

where * denotes the complex conjugate, and N is an estimate of the noise in the image. For this algorithm, *N* represents the variability between images. In order for such algorithms to allow for variability in the input image, the token must be robustly encoded, using some sort of error correcting code (ECC). Soutar et al. use a simple ECC based on Hamming distances and majority decision. For each bit of the secret to be encoded, a selection of locations in *R* with the same phase are written to a *link table*. For example, to encode a 1-bit, the *link table* may point to five values with a phase of +π/2. The template is created containing for following information: H_0, the *link table*, a cryptographic hash of the secret, and an identifier.

During *key release*, a new image F_1 is acquired. This image is deconvolved with the filter H_0, and the sign of the imaginary component taken as the estimate of the random function. We represent this as the calculation of R_1, as

$$R_0^* = sign\ (\ imag\ (\ H_0F_1\ )\ )$$

If F_1 is from the same individual as F_0, then R_1 should be a good estimate of R_0. This value is then used to decode the bits of the secret. Even if some link table values are incorrect, the use of *majority decision* should allow the correct value of the secret to be obtained.

4.0 RESULTS

In order to apply the attack described, it is necessary to create a match score from the template. For the biometric encryption system of Soutar et al. (1998) this is relatively straightforward. The number of elements of R_1 in each entry in the *link table* is an indicator of whether R_1 matches R_0, which results in a release of the secret code. For a random image f_1, approximately half of the elements in each link table entry will match, while for an identical image, all entries will match. Thus, the match score *MS* was calculated as sum of the number of elements of R_1 of the majority value – the number of elements of the minority value for each bit.

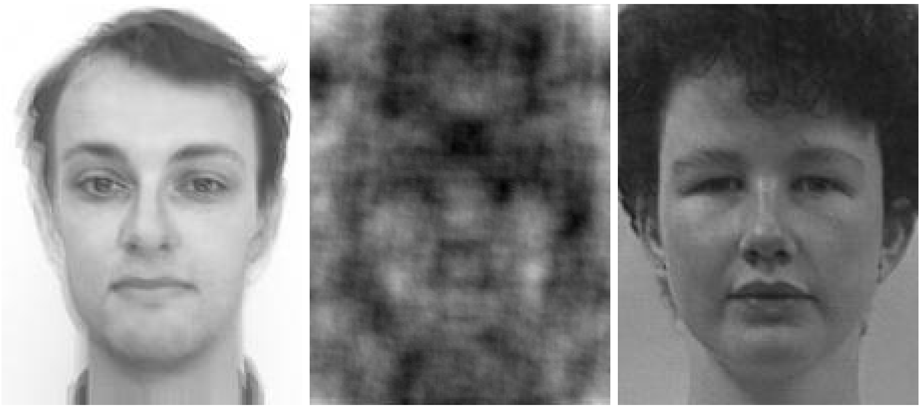

Figure 2: Sample images for an implementation of the biometric encryption technique of Soutar et al. (1998) for a face recognition. *Left*: Image F_0 averaged from ten samples. *Middle*: Template H_0 (represented as an image) including the random phase encoded elements. *Right*: Initial test image IM_0, chosen to be dissimilar to F_0.

In order to test this approach, we implemented a modified version of the algorithm Soutar et al. (1998) to apply to face recognition images. The advantage of this implementation is that the framework developed for hill-climbing for face recognition (Adler, 2004) would be applicable. On the other hand, such an algorithm is not realistic. Because face recognition data is not very distinctive, it would not be possible to encode many bits of a key (our initial results would suggest a maximum of about 20 bits). A template was created using 10 images from the University of Aberdeen face recognition database (Craw et al., 1999), and 20 secret bits were encoded using a link table with 5 elements of R_0 for each bit. In order to illustrate the power of the algorithm, an *initial image* intentionally different from the template was chosen. Figure 2 shows and image of the averaged enrolment images from the template (F_0), the encoded template (H_0), and the initial test image IM_k All images were scaled and rotated to have common size and eye locations.

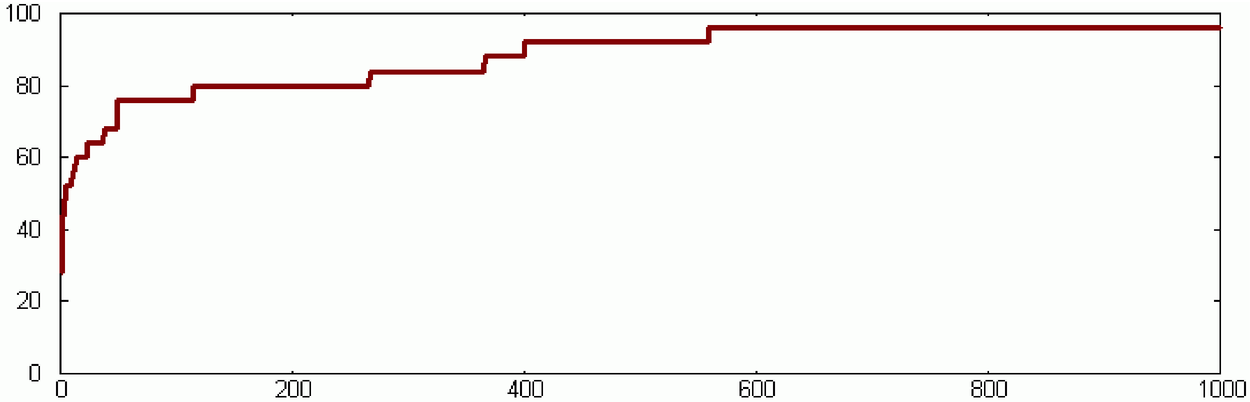

Figure 3: Match score *MS* versus iteration number. The match score is calculated as the number of bit positions matching in the template. A *MS* of 100 indicates a perfect match, while a value of 20 (1/5 of 100) would be expected for a random image.

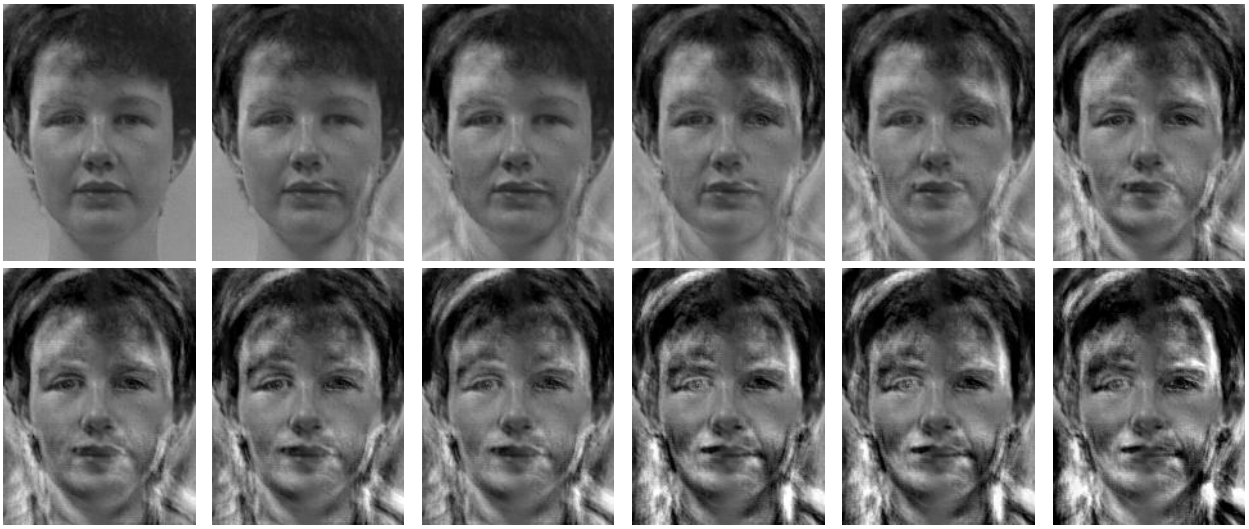

Figure 4: Sample images of IM_k for various iteration steps (from top left across to bottom right).

Results show that the template recreation algorithm is quickly able to attain an accurate match to F_0, with correct values for 96% of bit positions in R_1. This is easily adequate to decrypt the secret, and significantly larger than match values for other images of the enrolled individual (which were typically accurate to 82-86 bit positions). Figure 3 shows the graph of *MS* versus iteration number, while figure 4 shows a representative set of images IM_k as the algorithm progresses. There is an initial rapid increase in *MS* after which the algorithm shows a more gradual trend. It is interesting to note that *IM* begins to show some similar features to F_0 as iteration progresses. For example, the position of eyebrows, and shape of eyes, nose and chin begin to show a resemblance. One interesting aspect is that the hill-climbing algorithm does not seem to terminate with a final good estimate of the template image. Perhaps biometric encryption allows several possible variants of the enrolled image to match.

5.0 DISCUSSION

This paper presents an approach to attack biometric encryption algorithms in order to extract the secret code with less than brute force effort. A successful result was obtained for a simplified version of the biometric encryption algorithm of Soutar et al (1998). Essentially, this attack requires that the some information be "leaked" from the biometric encryption algorithm for sample images far from a match. This leaked information is used to construct a match score, which is subsequently used to iteratively improve an estimate.

While this work was implemented against the fingerprint algorithm of Soutar et al. (1998), more recent systems have been proposed, which appear to be somewhat less susceptible to this vulnerability. For example, the fingerprint algorithm of Clancy et al. (2003), encodes the secret as the coefficients of a Galois field polynomial. Minutiae points are encoded as pairs (x_i, y_i) where x_i is a minutiae point, and y_i is a point on the polynomial. Additionally, numerous "chaff" points are encoded, in which the value of y_i is random. During key release, the minutiae of the new fingerprint image are calculated, and the points x_i closest to the minutiae are chosen. The y_i corresponding to these points are used to estimate the polynomial, using a Reed-Solomon error correcting code framework. If enough legitimate points are taken, the correct polynomial will be obtained and the correct secret decrypted. This encryption technique is based on the "fuzzy vault" technique of Juels and Sudan (2002). An illustration of this scheme is shown in figure 5.

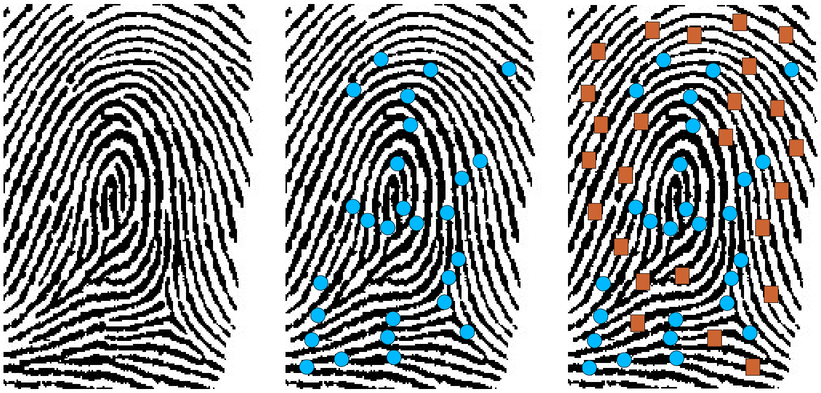

Figure 5: Schematic diagram of the biometric encryption scheme of Clancy et al. (2003). *Left* a raw fingerprint image is enrolled. *Middle* minutiae points (circles) are used to encode the value of a polynomial representing the secret. *Right* chaff points (squares), sufficiently far from minutiae, are used to encode random values of the polynomial.

We believe that it may be possible to use the attacks of this paper against the biometric encryption technique of Clancy et al. (2003), even though Juels and Sudan were able to give a proof of security. A key assumption for security proof is that the data held in the "fuzzy vault" are random. The data of Clancy, however, are not. Firstly, biometric data is inherently structured - otherwise hill-climbing wouldn't be possible. Secondly, the need to carefully place chaff minutiae points sufficiently far from legitimate ones is another source of non-randomness. However, at this time, we are not able to demonstrate an attack against this technique.

In their analysis, Uludag et al. (2004) note that most proposed biometric encryption systems only appear to account for a "limited amount of variability in the biometric representation." In order to quantify this notion, experiments were conducted by them to estimate the variability in fingerprint minutiae. Matched fingerprint pairs were imaged and minutiae locations identified by a human expert, which was assumed to give an upper bound on system performance. Using these data, the algorithm of Clancy et al. (2003) was analysed to estimate the FMR / FNMR tradeoff curve during key generation and key release. Results were surprisingly poor; an equal error rate of 6.3% can be estimated from the results, although the authors note that there are a limited number of feasible operating points. Furthermore, Uluday et al. (2004) note that

> "… these systems can, in practice, be defeated using relatively simple strategies. A naive attack on a biometric system could be launched by successively presenting biometric samples from a representative population (either synthetically generated or actual samples) and the success of the attack is likely to be bounded by the weakest link in the security chain, i.e., operating point of the biometric matcher."

In conclusion, this paper has presented a scheme that appears to show vulnerabilities in biometric encryption systems. The attacker estimates the enrolled biometric image and uses it to release the stored secret. The attacker considered here, who has access to biometric templates and authentication software, is quite plausible, as such biometric templates are typically stored in standardized formats to permit exchange between authorities, and may often be stored on identity documents, or otherwise by available.

6.0 REFERENCES

Adler A (2004) "Images can be regenerated from quantized biometric match score data" *Proc. Can. Conf. Elec. Comp. Eng.*

Adler A (2003) "Sample images can be independently restored from face recognition templates" *Proc. Can. Conf. Elec. Comp. Eng.* pp. 1163-1166

BioAPI Consortium (2001) *BioAPI Specification* http://www.bioapi.org/BIOAPI1.1.pdf 1163 -1166

Clancy T C, Kiyavash N, Lin D J (2003) "Secure smartcard-based fingerprint authentication" *Proc. ACMSIGMM 2003 Multimedia, Biometrics Methods and Applications Workshop* pp. 45-52.

Craw I, Costen N P, Kato T, Akamatsu S, (1999) How should we represent faces for automatic recognition? *IEEE Trans. Pattern Analysis and Machine Int.,* 21:725-736

Davida G I, Frankel Y, Matt B J (1998) "On enabling secure applications through off-line biometric identification" *Proc. IEEE Symp. Privacy and Security* pp. 148-157.

Davida G I, Frankel Y, Matt B J, Peralta R (1999) "On the relation of error correction and cryptography to an offline biometric based identification scheme" *Proc. Conf. Workshop Coding and Cryptography (WCC'99),* pp. 129-138.

Grother P, (2000) "Software Tools for an Eigenface Implementation" *National Institute of Standards and Technology,* http://www.nist.gov/humanid/feret/

Hill C J (2001), *Risk of Masquerade Arising from the Storage of Biometrics* B.S. Thesis, Australian National University, http://chris.fornax.net/biometrics.html

Juels A, Sudan M (2002), "A fuzzy vault scheme" *Proc. IEEE Int. Symp. Information Theory* p. 408.

Phillips P J, Moon H, Rauss P J, Rizvi S, (2000) "The FERET evaluation methodology for face recognition algorithms" *IEEE Trans. Pat. Analysis Machine Int.,* 22:1090-1104

Turk M A, Pentland A P, (1991) "Eigenfaces for recognition" *J. Cognitive Neuroscience,* 3:71-86

Uludag U, Pankanti S, Prabhakar S, Jain A K (2004) "Biometric Cryptosystems: Issues and Challenges", *Proc. IEEE,* 92(6): 948-960.

Uludag U, "Finger minutiae attack system" (2004b) *Proc. Biometrics Conference*