Secure Iris Recognition Based on Local Intensity Variations*

Christian Rathgeb and Andreas Uhl

University of Salzburg, Department of Computer Sciences, A-5020 Salzburg, Austria
{crathgeb,uhl}@cosy.sbg.ac.at

Abstract. In this paper we propose a fast and efficient iris recognition algorithm which makes use of local intensity variations in iris textures. The presented system provides fully revocable biometric templates suppressing any loss of recognition performance.

1 Introduction

Over the past years plenty of biometric traits have been established to be suitable for personal identification [1,2], iris being one of the most reliable [3]. Several iris recognition algorithms have been proposed throughout literature, reporting impressive recognition rates of over 99% and EERs below 1% on diverse datasets. However, iris recognition algorithms are still left to be improved with respect to computational performance as well as template protection, which has recently become an important issue [4,5]. Elapsed time during matching becomes relevant if huge databases are introduced whereas template protection guards users from identity theft. Fig. 1 shows a diagram of a generic iris recognition system.

The contribution of this work is the proposal of a new, computationally fast iris recognition algorithm providing practical recognition rates. By examining local intensity variations in preprocessed iris textures, features are extracted. We demonstrate the efficiency of our algorithm through recognition rates as well as comparing time measurements to a well-established algorithm. Furthermore, fully revocable templates are generated, meeting demands of high security applications. Revocable templates are created without the loss of recognition performance, while in many schemes, degradation of accuracy is observed [6].

This paper is organized as follows: first related work regarding iris recognition is summarized (Sect. 2). Subsequently the proposed system is described in detail (Sect. 3) and experimental results are given (Sect. 4). The security of our algorithm is discussed and a technique for providing secure revocable templates is proposed (Sect. 5). Finally, a conclusion is given (Sect. 6).

2 Related Work

Pioneer work in iris recognition was proposed by Daugman [7,8]. Daugman's algorithm forms the basis of today's commercially used iris recognition systems.

* This work has been supported by the Austrian Science Fund, project no. L554-N15.

A. Campilho and M. Kamel (Eds.): ICIAR 2010, Part II, LNCS 6112, pp. 266–275, 2010.

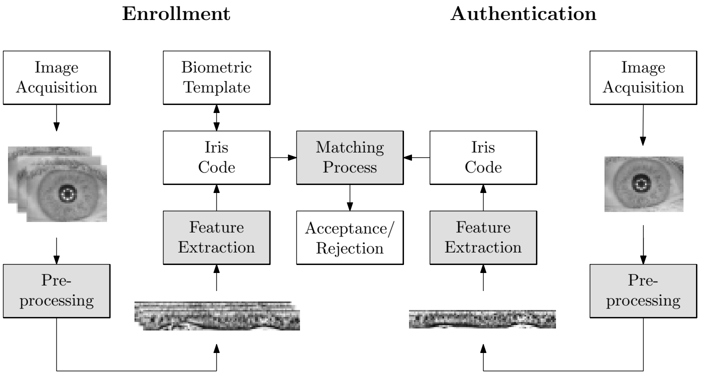

Fig. 1. Iris Recognition System: the common operation mode of enrollment and authentication in an iris recognition system

Within Daugman's approach each point of a preprocessed iris texture is treated as center of a 2D Gabor wavelet. For each of these wavelets the coefficients are generated out of which two bits are extracted resulting in an iris-code of a total number of 2048 bits. The matching process is performed using the Hamming distance as metric, comparing the number of mismatching bits against a threshold reaching an almost perfect recognition rate. A different approach to that presented by Daugman was proposed by Wildes [9]. Here an isotropic bandpass decomposition derived from application of Laplacian of Gaussian filters is applied to the preprocessed image data at multiple scales. That is, filtered images are realized as a Laplacian pyramid to generate the biometric template. In the matching process normalized correlation between acquired samples and stored template is calculated. Since these two first algorithms several approaches have been proposed suggesting several different filters to be used in the feature extraction step. Ma *et al.* [10] as well as Masek [11] examine 1D intensity signals applying a dyadic wavelet transform and a Log-Gabor filter, respectively. Chenhong and Zhaoyang [12] and Chou *et al.* [13] convolve iris images with a Laplacian-of-Gaussian filter. Ko *et al.* [14] apply cumulative sum based change analysis where iris textures are divided into cells out of which mean gray scale values are calculated and furthermore, upward and downward slopes of grayscale values are detected.

Approaches to template protection regarding iris biometrics have been proposed in so-called *Biometric Cryptosystems* [5]. Davida *et al.* [15] were the first to create a so-called "private template scheme" in which a hashed value of preprocessed iris codes and user specific attributes serves as a cryptographic key. By introducing error correcting check bits the scheme is capable of regenerating the hash at the time of authentication. Jules and Wattenberg [16] introduced a novel cryptographic primitive termed "fuzzy commitment scheme" which they suggest to be used in biometric cryptosystems. The key idea is to bind a cryptographic key prepared with error correcting codes with biometric data in a secure template. Additionally, a hash of the key is stored together with the template.

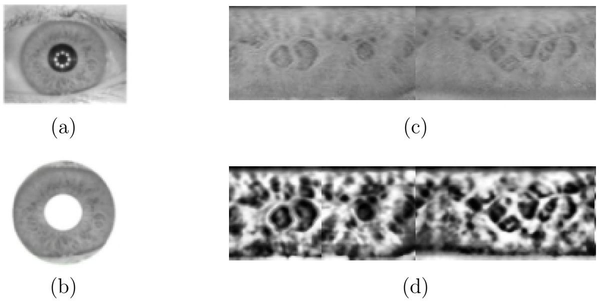

Fig. 2. Preprocessing: (a) an image of a person's eye is acquired (b) the iris is located and extracted (c) the iris ring is unwrapped to create a normalized iris texture (d) Gaussian blur and CLAHE contrast enhancement technique are applied to obtain a well distributed image

During authentication biometric data which is "close enough" (to some specified metric) to that captured during enrollment is able to reconstruct the key with the use or error correction decoding. The resulting key is then hashed and tested against the previously stored hash. Several systems applying the above concepts have been proposed. Although the main target of these schemes is biometric key management these techniques provide template protection as well [4]. Focusing on reported performance, in general security is increased at the cost of recognition rates. Additionally, iris-based cryptosystems are mostly based on existing iris recognition algorithms performing non-trivial feature extraction. Thus, performance with respect to runtime remains an issue.

In summary, most iris recognition systems exhibit high recognition rates, while these lag the requirement of providing secure biometric templates. Additionally, some algorithms are rather slow due to complex feature extraction techniques. Template protection schemes, such as biometric cryptosystems, provide secure templates, yet, security is mostly achieved at the cost of recognition performance.

3 System Architecture

3.1 Preprocessing

Preprocessing corresponds to the approach presented by Daugman [8]. Having detected the pupil of an eye, the inner and outer boundary of the iris are approximated. Subsequently, pixels of the resulting iris ring are mapped from polar coordinates to cartesian coordinates to generate a normalized rectangular iris texture. Due to the fact that the top and bottom of the iris are often hidden by eyelashes or eyelids, these parts of the iris are discarded (315^o to 45^o and 135^o to 225^o). To obtain a smooth image a Gaussian blur is applied to the resulting iris texture. To enhance the contrast we use an advanced contrast enhancement technique called CLAHE (Contrast Limited Adaptive Histogram Equalization) [17]. Compared to other contrast enhancement algorithms, for example histogram equalization, this algorithm operates on local image regions.

For this purpose the image is subdivided into image tiles (so-called contextual regions) and the contrast is enhanced within each of these regions. To avoid artifacts between two adjacent tiles an interpolation algorithm is employed. In Fig. 2 the entire preprocessing procedure is illustrated.

3.2 Feature Extraction

The applied feature extraction technique represents the fundamental part of our system. By tracing intensity variations in horizontal stripes of distinct height of preprocessed iris textures, so-called “pixel-paths” are extracted. We found that these paths are suitable to identify users.

First of all, the preprocessed iris texture of a person i, I_i (in form of a rectangle), is divided into n different horizontal texture stripes

$$I_i \rightarrow \{I_{i1}, I_{i2}, ..., I_{in}\} \tag{1}$$

of height h pixels (needless to say n depends on the size of h). Each texture strip is of dimension $l \times h$, where l denotes the length of preprocessed iris textures.

In the next step two pixel-paths, representing light and dark intensity variations are created for each texture strip I_{ij}. We define these paths as,

$$P_{Lij} := \{p_{Lij0}, p_{Lij1}, ..., p_{Lijl}\} \tag{2}$$

$$P_{Dij} := \{p_{Dij0}, p_{Dij1}, ..., p_{Dijl}\} \tag{3}$$

To calculate the value of elements p_{Lijk} and p_{Dijk} of each of these paths the first element of each path is defined as

$$p_{Lij0} \leftarrow h/2, \qquad p_{Dij0} \leftarrow h/2 \tag{4}$$

In other words, each path starts at the leftmost center of the according strip. Elements p_{Lij1} and p_{Dij1} are then calculated by examining the three directly neighboring pixel values of p_{Lij0} and p_{Dij0} ($p_{Lij0} = p_{Dij0}$) in next pixel column. Then p_{Lij1} is set to the y-value of the maximum and p_{Dij1} is set to the y-value of the minimum of these three values (maxima and minima correspond to brightest and darkest grayscale values of pixels). Thus, we define the values of p_{Lijk} and p_{Dijk} recursively such that,

$$L := MAX \begin{pmatrix} I_{ij}[k+1, p_{Lijk}-1], \\ I_{ij}[k+1, p_{Lijk}], \\ I_{ij}[k+1, p_{Lijk}+1] \end{pmatrix} \tag{5}$$

$$D := MIN \begin{pmatrix} I_{ij}[k+1, p_{Dijk}-1], \\ I_{ij}[k+1, p_{Dijk}], \\ I_{ij}[k+1, p_{Dijk}+1] \end{pmatrix} \tag{6}$$

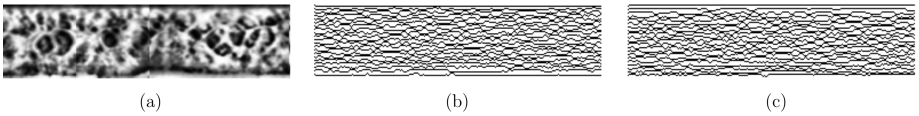

Fig. 3. Feature Extraction: (a) the preprocessed iris texture (b)-(c) two pixel-paths are extracted for each texture strip respresenting light and dark intensity variations for a height of $h = 3$ (notice that the paths of light and dark intensity are not necessarily complementary)

where $I_{ij}[x, y]$ represents the pixel value of the jth texture strip at coordinates $[x, y]$ and the MAX and MIN functions derive the according points. The values of p_{Lijk+1} and p_{Dijk+1} are then set to the y-values of L and D:

$$p_{Lijk+1} \leftarrow L_y \tag{7}$$

$$p_{Dijk+1} \leftarrow D_y \tag{8}$$

This means, $p_{Lijk} \in \{0, 1, ..., h-1\}$ and $p_{Dijk} \in \{0, 1, ..., h-1\}$. In the case p_L or p_D reach the top or bottom of the texture strip, only the according two directly neighboring pixel values are taken into account. An example for constructing light and dark intensity paths is shown in Fig. 3.

To complete the feature extraction extracted paths are further smoothed, which means small peaks are discarded. For this purpose a threshold t is defined and variations of y-position of pixel-paths occurring within a range of t pixels are discarded in order to smooth the whole path. An example of smoothing pixel-paths is illustrated in Fig. 4. The top and the bottom strip are discared since we found that those stripes normally do not carry useful information.

As a result of the described feature extraction procedure extracted paths are stored for the ith user. The size of the generated template depends on the size of the preprocessed iris texture as well as parameter h. For a number of n stripes $2 \times n \times l$ elements out of the set $\{0, 1, ..., h-1\}$ form the biometric template. At the time of enrollment, where a user i registers with the system, feature extraction is performed for a single iris image and a biometric template T_i is stored. In Sect. 5 we will discuss how to secure these templates.

The feature extraction is based on simple comparisons, thus, no complex calculations are required. With respect to systems where computational simplicity and runtime of feature extraction are issues (for example, smart-card based verification systems) the proposed feature extraction method provides fast computation based on simple comparisons.

3.3 Template Matching

For the ith user, the feature extraction generates a template, denoted by T_i. This template consists of $2 \times n \times l$ integer values which correspond to y-positions of elements of extracted pixel-paths for light and dark intensity variations. To calculate the similarity between two templates T_j and T_k the square of differences of all elements of T_j and T_k are summed up in a matching value M such that

Fig. 4. Smoothing Pixel-Paths: (a) the original pixel-path resulting out of a texture stripe (b) the smoothed pixel-path where t is set to 4. That is, all peaks in the y-direction lying within a range smaller than 4 are discarded.

$$M_{jk} := \sum_{m=0}^{N} (abs(T_{jm} - T_{km}))^2 \tag{9}$$

where N is set to $2 \times n \times l$. By definition small differences increase the matching value slightly, large differences increase the matching value significantly. A small matching value indicates high similarity between templates and vice versa. Depending on the chosen size of h the highest possible match value varies. An appropriate threshold has to be set up according to intra-class and inter-class distributions of genuine and non-genuine users.

In comparison to existing approaches aiming at extracting distinct binary iris-codes which are matched by comparing the Hamming distances of two iris-codes against a predefined threshold (for example, [8,9,11,10]), the proposed matching process lags performance and, thus, is expected to be inappropriate for template matching on large-scale databases. To overcome this restriction we introduce a more efficient way of matching templates for an appropriate height h in Sect. 4.2.

4 Experimental Results

Experiments are carried out using the CASIAv3-Interval [18] iris database, which comprises iris images over two-hundred different persons, where on average about 6 iris images are available per person. As a result of the preprocessing procedure iris textures of 256×64 pixels are extracted ($\Rightarrow l = 256$). A total number of $2 \times n \times 256$ integers are extracted and stored as biometric template. For each person a single iris image is processed in the enrollment step. Additionally, a circular shift of ten pixels to the left and to the right is implemented in order to provide rotation invariance for small head tilts.

4.1 Recognition Performance

In our experiments best results were obtained for stripes of height $h = 3$ pixels. For a height of $h = 3$ we get $64/3 = 21$ stripes where the top and bottom strip are discarded, resulting in a total number of 19 stripes. Each strip consists of 256 integers in the range $[0, 3)$. That is, an iris-code of $256 \times 19 \times 2 = 9728$ codewords out of the set $\{0, 1, 2\}$ is stored in the template. Experimental results for several different values of h are summarized in Table 1, where for all values of h best results were achieved with a threshold of $t = 4$.

The false rejection rate (FRR) and false acceptance rate (FAR) for a height of $h = 3$ and a threshold of $t = 4$ are plotted in Fig. 5. For zero FAR a FRR of

Table 1. Performance measurements for the proposed systems according to different values of h for a threshold of $t = 4$ and recognition rates of existing algorithms

Height (Pixels) / Algorithm	*FRR(%) @ FAR = 0*	*EER (%)*
2	3.829	2.227
3	1.978	1.016
4	3.959	2.128
5	5.817	2.992
Masek [11]	3.952	2.477
Ma *et al.* [10]	1.817	1.073
Ko *et al.* [14]	20.531	4.738

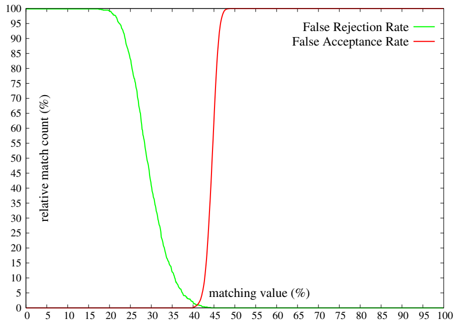

Fig. 5. The false rejection rate and the false acceptance rate of the proposed algorithm for a height of $h = 3$ and a threshold of $t = 4$

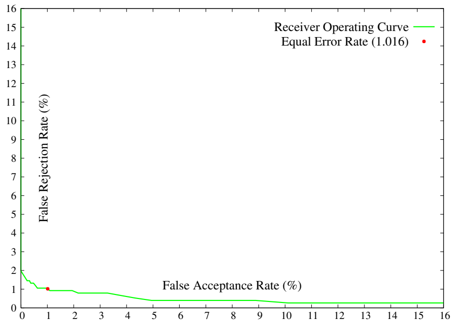

Fig. 6. The receiver operating curve and the equal error rate of the proposed algorithm for a height of $h = 3$ and a threshold of $t = 4$

1.978% is obtained. The according receiver operating curve is plotted in Fig. 6 resulting in an EER of 1.016%. Compared to our own implementations of existing iris recognition algorithms (see Table 1), these are satisfying results with respect to the simplicity of the proposed feature extraction method.

4.2 Computational Performance

The above described system was implemented in C and tested on a 1.3 GHz Linux system. As mentioned earlier the feature extraction method is based on simple comparisons between grayscale values. In detail, the maximum and minimum of three numbers are calculated using three comparisons. That is, for a height of $h = 3$ a total number of $256 \times 19 \times 3 = 14592$ comparisons are necessary. Measuring the runtime of the feature extraction method for a single preprocessed iris texture an average processing time of 0.0344 seconds was obtained. To emphasize the performance of the feature extraction method we compare the computational performance to our C implementation of the algorithm of Ma *et al.* [10]. In the algorithm of Ma, a 1-D wavelet transform is applied to ten 1-D intensity signals of average grayscale values of pixel blocks in the preprocessed iris texture. Detected minima and maxima serve as features where sequences of 1 and 0 are assigned to the iris-code until new maxima or minima are found. This whole process is applied to two subbands extracting a total number of 10240 bits where the Hamming distance is applied as similarity metric. As experimental results of our implementation of this algorithm we achieved a FRR of 3.821% for zero FAR and a EER of 1.401% for the whole CASIAv3-Interval [18] iris database (circular shifts are implemented as well; we do not consider any bit-masking information). We measured the runtime of the feature extraction of this algorithm on the same system resulting in an average processing time of 0.1345 seconds for a single feature extraction. On average, the proposed feature extraction method is three times faster than that of Ma.

While most iris recognition systems compare iris-codes by calculating the Hamming distance between these, the matching procedure of our algorithm involves the calculation of a matching value which is expected to be much slower. Since we obtained best results for a height of $h = 3$ we are able to gain performance. To retrieve binary iris-codes we encode elements of calculated pixel paths with a Gray code:

$$0 \leftarrow 00, \qquad 1 \leftarrow 01, \qquad 2 \leftarrow 11 \tag{10}$$

By applying this encoding, calculating the Hamming distance between two iris-code generates the same results as the previously described matching. For a height of $h = 3$ the matching process is now computationally efficient as well. Compared to the algorithm of Ma which extracts 10240 bits we extract a total number of $2 \times 9728 = 19456$ bits. However, calculating the Hamming distance for larger bitstreams does not drastically decrease performance. For the algorithm of Ma we measured an avergage time of 0.0137 seconds and for the proposed we now achieve a average time of 0.0193 seconds for the matching of two templates. The time for calculating the Hamming distance between two bitstreams of twice the length of those generated by the algorithm of Ma takes only slightly longer, due to system overhead.

5 Cancellable Templates

Recently template security has become an important issue [4]. If biometric templates are stolen or compromised these can not be modified ex post and, thus, become useless. Ratha *et al.* [6] introduced the concept of cancellable biometrics. The idea of cancellable biometrics consists of intentional, repeatable distortion on a biometric signal based on a chosen transform where the matching process is performed in transformed space. Recovering of original biometric template data becomes infeasible. If the transformed biometric data is compromised the transform function is changed, that is, the biometric template is updated.

In order to provide cancellable biometric templates we suggest a permutation of extracted paths following the idea of line permutation as proposed by [19]. For example, for a height of $h = 3$ we calculate a total number of 38 paths.

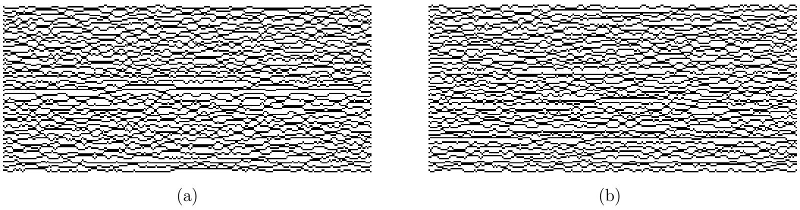

Fig. 7. Secure Template Creation: (a) paths calculated during feature extraction (b) secure template defined by a distinct permutation

Paths are permutated according to some user or application specific permutation, where a total number of $38! \simeq 5.23 \cdot 10^{44}$ different permutations are possible. By permutating paths, reconstructing original iris images becomes highly complicated. Guessing a specific permutation is assumed to be computationally infeasible ($38! \gg 2^{128}$). Thus, high security regarding template protection is provided. In Fig. 7 a sample invertible permutation is illustrated. If non-invertible permutations are applied in our system, performance decrease is expected as pointed out in [19]. In case a specific permutation is compromised, an imposter may reconstruct the original order of paths. However, reconstructing the original iris texture from iris codes is not possible, since the feature extraction is non-invertible by definition.

By introducing a two-factor authentication scheme permutations are integrated in the system, where secret permutations represent the second factor. For example, user-specific permutations could be stored on smart-cards, so that permutations are applied after feature extraction and permutated templates are matched against stored templates, previously permuted during enrollment. In comparison to template encryption our system is capable of performing the matching procedure in the encrypted (permuted) domain. Furthermore, compared to approaches to cancellable iris biometrics which operate in the image domain [20], the proposed system does not suffer from performance degradation if invertible permutations are applied. This is one important aspect of the presented approach since security applications must not require a decryption of encrypted templates prior to matching [4].

6 Conclusion

In this work we presented a new, computationally efficient iris recognition algorithm. Besides providing practical recognition rates we demonstrate that the proposed algorithm is suitable for generating secure and fully revocable biometric templates.

Acknowledgements

We would like to thank Michael Liedlgruber for providing us with an implementation of the CLAHE contrast enhancement technique.

References

1. Jain, A.K., Ross, A., Prabhakar, S.: An introduction to biometric recognition. IEEE Trans. on Circuits and Systems for Video Technology 14, 4–20 (2004)
2. Jain, A.K., Flynn, P.J., Ross, A.A.: Handbook of Biometrics. Springer, Heidelberg (2008)
3. Bowyer, K., Hollingsworth, K., Flynn, P.: Image understanding for iris biometrics: a survey. Computer Vision and Image Understanding 110, 281–307 (2008)
4. Jain, A.K., Nandakumar, K., Nagar, A.: Biometric template security. EURASIP J. Adv. Signal Process., 1–17 (2008)
5. Uludag, U., Pankanti, S., Prabhakar, S., Jain, A.K.: Biometric cryptosystems: issues and challenges. Proceedings of the IEEE 92(6), 948–960 (2004)
6. Ratha, N.K., Connell, J.H., Bolle, R.M.: Enhancing security and privacy in biometrics-based authentication systems. IBM Systems Journal 40, 614–634 (2001)
7. Daugman, J.: High confidence visual recognition of persons by a test of statistical independence. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(11), 1148–1161 (1993)
8. Daugman, J.: How Iris Recognition Works. IEEE Trans. CSVT 14(1), 21–30 (2004)
9. Wildes, R.P.: Iris recognition: an emerging biometric technology. Proceedings of the IEEE 85, 1348–1363 (1997)
10. Ma, L., Tan, T., Wang, Y., Zhang, D.: Efficient Iris Recognition by Characterizing Key Local Variations. IEEE Transactions on Image Processing 13(6), 739–750 (2004)
11. Masek, L.: Recognition of Human Iris Patterns for Biometric Identification, Master's thesis, University of Western Australia (2003)
12. Chenhong, L., Zhaoyang, L.: Effcient iris recognition by computing discriminable textons. In: Interantional Conference on Neural Networks and Brain, vol. 2, pp. 1164–1167 (2005)
13. Chou, C.T., Shih, S.W., Chen, W.S., Cheng, V.W.: Iris recognition with multiscale edge-type matching. In: Interantional Conference on Pattern Recognition, pp. 545–548 (2006)
14. Ko, J.G., Gil, Y.H., Yoo, J.H.: Iris Recognition using Cumulative SUM based Change Analysis. In: Intelligent Signal Processing and Communications, ISPACS'06, pp. 275–278 (2006)
15. Davida, G., Frankel, Y., Matt, B.: On enabling secure applications through off-line biometric identification. In: Proc. of IEEE, Symp. on Security and Privacy, pp. 148–157 (1998)
16. Juels, A., Wattenberg, M.: A fuzzy commitment scheme. In: Sixth ACM Conference on Computer and Communications Security, pp. 28–36 (1999)
17. Zuiderveld, K.: Contrast Limited Adaptive Histogram Equalization. In: Graphics Gems IV, pp. 474–485. Morgan Kaufmann, San Francisco (1994)
18. The Center of Biometrics and Security Research: CASIA Iris Image Database, `http://www.sinobiometrics.com`
19. Zuo, J., Ratha, N.K., Connel, J.H.: Cancelable iris biometric. In: Proceedings of the 19th International Conference on Pattern Recognition (ICPR'08), pp. 1–4 (2008)
20. Hämmerle-Uhl, J., Pschernig, E., Uhl, A.: Cancelable iris biometrics using block remapping and image warping. In: Samarati, P., Yung, M., Martinelli, F., Ardagna, C.A. (eds.) ISC 2009. LNCS, vol. 5735, pp. 135–142. Springer, Heidelberg (2009)