Pattern Recognition Letters 33 (2012) 1006–1011

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Noisy Iris Recognition Integrated Scheme

Maria De Marsico [a], Michele Nappi [b], Daniel Riccio [b,*]

[a] Sapienza Università di Roma, Via Salaria 13, 00198 Roma, Italy
[b] Università di Salerno, via Ponte don Melillo, 84084 Fisciano (SA), Italy

ARTICLE INFO

Article history:
Available online 16 September 2011

Keywords:
Iris recognition
LBP
BLOB

ABSTRACT

One of the most challenging issues in iris recognition is the design of techniques able to ensure high accuracy even in adverse conditions. This paper deals with an approach to iris matching based on the combination of local features: Linear Binary Patterns (LBP) and discriminable textons (BLOBs) are presently exploited. The techniques have been refined ad hoc, to allow the extraction of significant discriminative features, even with images captured in variable visible light conditions, and affected by noise due to distance/resolution or to scarce user collaboration (blurring, off-axis iris, occlusion by eyelashes and eyelids). The obtained results strongly motivate further investigations along this line, most of all the addition of more local features.

1. Introduction

Iris is in the list of the most reliable biometrics which are presently used or investigated. In fact, it is able to provide a comparable or even higher accuracy than other biometrics, e.g. fingerprints. However, this is only possible by ensuring strong constraints for subject cooperation and quality of the captured image. Therefore, present research trend is towards high accuracy, though trying to significantly relax the constraints. The main goal is a matching system with a sufficient robustness with respect to different kinds of distortion, such as blurring, off-axis, reflections and occlusions due to eyelids or eyelashes. As a matter of fact, in a semi-controlled setting, due either to lower user's cooperation, or to limited performances of the capture device, the system must work over noisy iris images, which are often partially compromised. To face this problem, we propose the Noisy Iris Recognition Integrated Scheme (N-IRIS). It adopts and combines two local feature extraction techniques, Linear Binary Patterns (LBP) and extraction of discriminable textons (BLOBs), which differently and independently characterize relevant regions of iris.

LBP produces a local texture descriptor by low computational cost, which is appropriate to analyze images with high resolution or in real-time, and is robust to monotone variations of gray levels, which is useful in applications such as iris recognition. In fact most approaches to iris recognition are based on statistical classifiers and local features.

While LBP identifies quite regular patterns, the uniqueness of the iris texture is generally characterized by the irregular distribution of local feature blocks such as *furrows*, *crypts* and *freckles* or *spots*. Such features can be considered as blobs: a group of image pixels which form a structure which can be darker or lighter than the surrounding region. The extraction of the blobs from an iris image is obtained through different LoG (Laplacian of Gaussian) filter banks. We will refer to this technique as BLOB.

After investigating possible adaptations of the single approaches to the case at hand, we combined LBP and BLOB, and based recognition on a weighted mean of matching results at score level (Fig. 1). Experimental results show that such combination of the two techniques, though not particularly complex, provides better accuracy results than those obtained from the single approaches. This suggests that different kinds of iris features may call for different suited codings for a better matching. Possible future studies will focus on the combination of more kinds of features, as well as the design of more sophisticated schemes for the integration of different information.

2. Image preprocessing and segmentation

The first processing phases of any iris-based identification system are iris location and segmentation. The precision of the separation between the useful region for identification and those that can be considered as noise elements (reflections, eyelids, eyelashes) is of paramount importance. The higher such precision, the more informative the obtained iris code, and therefore the better the expected recognition result.

Two main regions can be identified inside the iris: the pupillary region, which is the innermost one and determines pupil's contour, and the ciliary region which is the outermost one and surrounds the pupillary region. These regions are separated by a third one called collarette. Further important elements which are taken into

* Corresponding author.
E-mail address: driccio@unisa.it (D. Riccio).

0167-8655/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2011.09.010

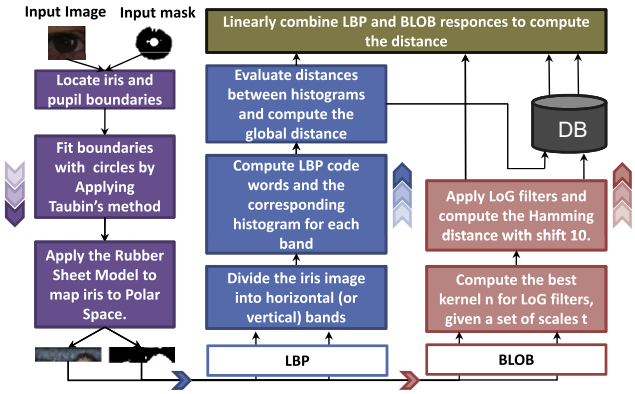

Fig. 1. The architecture of N-IRIS.

account during segmentation as well as coding are sclera, eyelids and eyelashes. These latter two may often hinder a correct segmentation, and may lead to a poor coding if they are included in the pupil code. On the other hand, useful structures for recognition are crypts, circular and radial furrows, freckles and spots with various extent.

Though strictly correlated, according to the preceding considerations, segmentation and matching represent two well distinguishable steps. As a matter of fact, also international challenges addressing iris recognition, e.g. NICE, have been divided into two different phases. NICE I explicitly and uniquely addressed the problem of segmenting noisy iris images, and the best results were obtained by the approach presented by Tan et al. (2010). On the other hand, NICE II uniquely focused on noisy irises matching, by preventively providing both iris images and the corresponding precomputed segmentation mask images, obtained by the algorithm presented by Tan et al. (2010) (Fig. 2 shows some examples). N-IRIS exploits such segmentation mask to refine and transform the iris region into a rectangular region, from which features are then extracted.

First of all, we approximate iris and pupil boundary by circumferences (centre and radius) as accurately as possible, so as to allow to pass from the image Cartesian space to the iris region polar space. Possible distortion introduced in this phase invalidate all the following steps. A naive approach would suggest to extract contours from objects in the mask and approximate them by circumferences, possibly solving an ellipse fitting problem. However, as Fig. 2 (b and c) shows, this problem may hide traps that a so trivial algorithm is not able to handle. The second column (b) of Fig. 2 demonstrates that ellipse fitting is too sensible to discontinuities introduced in the iris and pupil contours by occlusions due to reflections or eyelids. Curves resulting from contour approximation tend to get completely deformed just to precisely adhere to the available boundary portion. A further problem arises in all those cases which are similar to the iris in the third row of Fig. 2; the black region contour represents a single object without discontinuities. This makes it difficult to distinguish pupil frontier points from pupil contour ones.

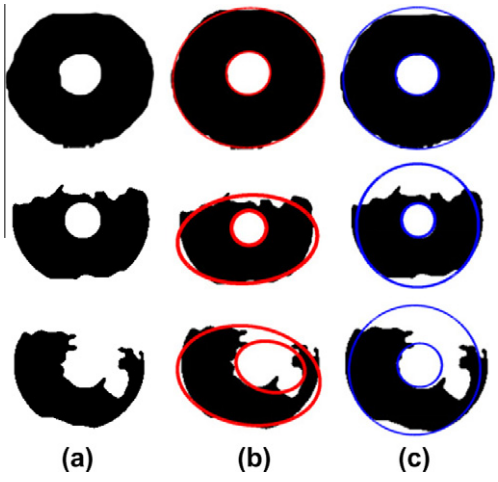

Fig. 2. Column (a) shows, from top to bottom, iris masks of increasing difficulty; column (b) shows mask processing by ellipse fitting; column (c) shows mask processing by N-IRIS.

N-IRIS adopts a more articulate solution. To address problems caused by images like that in the third row of Fig. 2, segmentation starts from identifying the pupil contour first, and proceeds by separating the pupil from the iris region. The mask is scanned row by row from top to bottom. Each row is scanned from the first to the last column, by marking the first and the last black pixels. These pixels represent the iris frontier. Frontier points undergo the algorithm by Taubin (1991) to approximate planar curves, surfaces and non planar space curves through implicit equations (more details on this method are discussed in the cited paper). Once the centre and radius of the circumference representing the iris are known, a further concentric circumference internal to it is considered, with radius equal to 1/5 of that of the iris. All pixels which fall outside such circle are deleted, and the procedure of circle fitting is repeated on this new image to determine the centre and the radius of the circumference which approximates the pupil.

Before proceeding to actual feature extraction, one needs to transform the iris region in a suitable form, also considering future matching operations. A first possible problem regards capture distance, so that the iris diameter may not be constant. This dimension must be normalized, yet avoiding to loose details or to introduce "ghost" information. Iris shape influences matching, therefore the chosen representation must provide for possible translations and rotations. Pupil can be differently dilated in different capture sessions, due to different lighting conditions, and this must be taken into account. Finally, pupil is seldom exactly located at the center of the iris. One has therefore to transform the iris so that the iris representation is constant in dimensions and that relevant features are approximately located in the same points. In this work we exploit the Rubber Sheet Model by Daugman (2004), which transforms the iris in radial coordinates while fixing the final dimensions of the obtained (rectangular) image. Due to anticipated scarce resolution of iris images at hand, we adopted a radial resolution (number of pixels along a radial line) of 40 pixels, and an angular resolution (number of radial lines around the iris region) of 360 pixels (see the implementation by Pigini (2010)). The same normalization is separately performed on the segmentation mask associated to each image. The mask is such that M (x,y)=1 if I (x,y) is a pixel of noise, and M (x,y)=0 otherwise, so that only information in relevant iris regions is coded (Fig. 3).

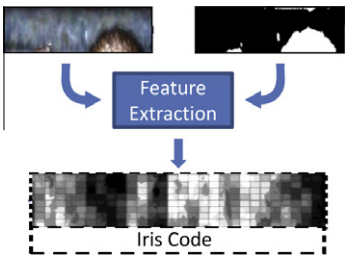

Fig. 3. Feature extraction and coding based on normalized iris image and segmentation mask.

3. Extracting local features

After pupil and iris contours identification, and after noisy elements have been accounted for (e.g. eyelashes, which occupy different positions and extent in different captures of the same subject), the relevant iris annulus must be coded. The different nature of its useful visual structures seems to call for different coding schemes, and for the fusion of their results as in multimodal feature extraction. The present work describes a preliminary phase of this process, where we exploited a version of LBP to record the textural regularities present in the iris, and blob identification for coding lighter or darker spots inside the iris region. Future developments will investigate the addition of appropriate versions of more local operators.

N-IRIS tries to merge the specific strengths of the two selected techniques. Our first attempts aimed at investigating the usefulness of local texture analysis based on Local Binary Pattern (LBP) [Ahonen et al. (2006), Ojala et al. (2002), Mäenpää et al. (2000)]. In particular, we evaluated the solution in Sun et al. (2006) and finally devised our own version, based on experimental evidence on the best strategy to be adopted. In order to further enhance the obtained results, we also investigated the combination of our implementation of LBP with the work by Chenhong and Zhaoyang (2005, 2008), that we refer to as BLOB. We implemented the described algorithm to extract discriminable textons, representing image regions which are lighter or darker than the surrounding zone, and also made some attempts to enhance it. Then, we merged the matching criteria stemming from the two techniques, to exploit the respective strengths.

3.1. Liner Binary Patterns – LBP

The Local Binary Pattern (LBP) was introduced by Timo Ojala and Harwood (1996) to analyze image texture. In its basic version, the operator associates to each pixel in the image a value, which is computed according to its 3×3 neighbourhood. This value is the decimal representation of the binary string (number) obtained by comparing the value of the pixel with each value in its neighbourhood. If the central pixel has a lower value than one of its neighbours, a 1 is recorded in the string for such neighbour, and a 0 otherwise (Fig. 4). A variation is presented by Ojala et al. (2002), where the basic operator is extended to process pixel neighbourhoods of variable dimension, and to be invariant to rotations. The circular neighbourhood of a pixel is exploited, and sample points are identified by interpolation. The resulting operator is called LPBP,R where P is the number of sample points, and R is the radius of the neighbourhood.

In the work by Sun et al. (2006), a further variation of LBP is used for iris recognition. The normalized iris image is divided into blocks. For each of them, an histogram is computed, and then a graph structure is created and stored for matching. We use a less computationally expensive solution by dividing the iris image into horizontal (or vertical) bands (Fig. 5). For each band, the histogram of LBP values is computed. The resulting code C will be stored as a sequence of histograms plus the related noise mask M: $C = (H_1, H_2, \ldots, H_{bands}, M)$. We assume that such a mask is always produced at the end of the image segmentation process. It is used during matching to take into account the amount of noise which is

Fig. 4. Computation of LBP.

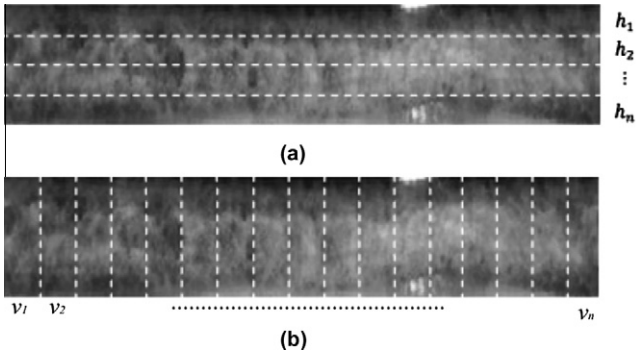

Fig. 5. Division in: (a) horizontal bands, (b) vertical bands.

present within the compared bands. The higher the number of noise pixels in the matched bands, the less reliable the similarity measure between the histograms.

Given two codings $C1 = (H_1, H_2, \ldots, H_{bands}, M)$ and $C2 = (K_1, K_2, \ldots, K_{bands}, N)$, and any histogram similarity measure (e.g. correlation, intersection or Bhattacharyya), matching is performed by computing the mean of the following values:

$$\delta(H_b, K_b)\left(1 - \frac{\overline{noise_b}}{totpixel}\right) \forall b \in \{1, \ldots, |bands|\} \quad (1)$$

where $\overline{noise_b}$ represents the mean number of noise pixels in the b-th band of masks M e N. Bands specialize blocks: a generic block is $m(rows) \times n(columns)$ pixels, an horizontal band is a $1 \times n$ block and a vertical band is a $m \times 1$ block. Once blocks are ordered row-first, formula (1) always holds.

Our most significant experiments with LBP on UBIRIS.v1 and UBIRIS.v2 are reported in Section 5. They aimed at testing both different types (block, vertical, horizontal) and numbers of bands; a division in five horizontal bands was optimal for both databases. We observe that the number of bands is mostly related to the normalization parameters. Once these are fixed, results are not affected by this choice. It is also interesting to notice that the most accurate solution in terms of type of bands is also the most bound to anatomical features, since horizontal bands in the polar image correspond to circular bands in the original image, and therefore are expected to be quite significant in coding iris features.

3.2. BLOB

Differential operators can address the problem of identifying lighter or darker regions in the iris. The combination of the Laplacian operator (effective as contour detector, but very sensible to noise) with a Gaussian filter (to preliminarily smooth the image) presents two advantages: noise reduction, due to smoothing, and better blob setting off, due to increased size of the Gaussian filter. This is the core idea of what we call BLOB, presented by Chenhong and Zhaoyang (2005, 2008), where iris blobs are modeled by a Gaussian 2-dimensional non-symmetric function, with length features $\sqrt{t_1}$ and $\sqrt{t_2}$:

$$f(x_1, x_2) = g(x_1; t_1)g(x_2; t_2) = \frac{1}{\sqrt{2\pi t_1}} e^{\frac{x_1^2}{2t_1}} \cdot \frac{1}{\sqrt{2\pi t_2}} e^{\frac{x_2^2}{2t_2}} \quad (2)$$

To identify blobs of different sizes, the representation must be given both in space and in scale. For the semi-group property of Gaussian kernels $g(\cdot; t_A) * g(\cdot; t_B) = g(\cdot; t_A + t_B)$ the authors derive:

$$L = g(x_1; t_1 + t)g(x_2; t_2 + t) \quad (3)$$

If an image undergoes a space-scale smoothing, values of spatial derivatives generally decrease with scale. Then it is necessary to

use a normalized differential operator ∇^2_{norm}. The authors show that the normalized response of a blob detector at scale t is:

$$\nabla^2_{norm} L = t(\nabla^2 L) \tag{4}$$

The solution by Chenhong and Zhaoyang (2005, 2008) to extract and code blob features is: fix the different scales, compute $\nabla^2_{norm} L$ for each scale (Fig. 6), and fuse the results by taking, for each pixel, the maximum value among all scales. Popular computational tricks allow to fuse Gaussian and Laplacian in a single LoG operator. Here the sizes of the convolution kernels at different scales were found using cross-validation, e.g. regression.

We obtain a matrix with both positive and negative values, where the former represent dark spots and the latter light ones. The result is binarized by setting all negative values to 0 and all positive values to 1. An example is in Fig. 6. Matching between two binary codes can be performed using Hamming distance weighted by the segmentation masks, as discussed by Daugman (2004). We also considered shifts of 10 pixels to address rotation variations. The final distance is the one computed on the alignment returning the maximum match. We also tested an alternative strategy, by chaining the separate scale (binary) codings in a longer code, instead of fusing them. Matching was performed by comparing codes at the same scale and taking the mean of obtained values as distance. We will refer to this modality as *chain*, as opposed to the original one (*fusion*). It seemed to rely on more discriminative information, but this did not produce the expected improvements.

3.3. LBP-BLOB

LBP-BLOB is the name we gave to the fusion of LBP and BLOB methods. The two codes are simply chained, and results from the two matching procedures are fused at score level, so that, at present, coding and matching of one method do not affect the other. Given I a normalized iris image and M its normalized segmentation mask, we call c_{LBP} and c_{BLOB} respectively the LBP and BLOB coding of the couple (I,M), which is performed only on the I element. We define the final method for coding and matching as:

- Coding of the pair (I,M) is $c = \{c_{LBP}, c_{BLOB}\}$
- Matching between codings c_1 and c_2 is given by:

$$\delta(c_1, c_2) = \frac{\delta_{LBP}(c_{1,LBP}, c_{2,LBP})}{2} + \frac{\delta_{BLOB}(c_{1,BLOB}, c_{2,BLOB})}{2} \tag{5}$$

Our fusion strategy was assessed by experiments on a larger set than the competition one (see details by Sammartino (2010). On this sufficiently substantial test bed, we observed that LBP and BLOB show a quite uncorrelated behaviour in terms of ability to discriminate between genuine and impostor matches. Though this is not a formal proof of the actual lack of correlation between the two techniques, it is an expected result if we consider that they rely on different theoretical frameworks, aiming to capture different relevant characteristics (texture regularity and the presence of significant "hot spots"). For the same reason, we can expect that one method works better than the other on specific images. In our future research lines, a related study represents a core point. For the time being, the previous observations can explain, in the present setting, why the simple sum of the two scores improves the performances of the single classifiers, as confirmed by experimental results.

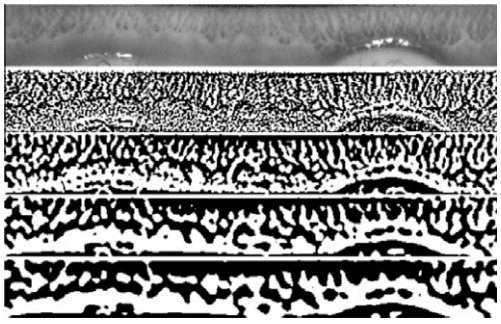

Fig. 6. Application of LoG filter at increasing scales (from Chenhong and Zhaoyang (2008)).

In a similar way, we have no systematic study on the score distributions produced by the two techniques. In a future work, we will further investigate such aspect. This might also allow to replace the simple score sum with a probabilistic framework, e.g. Bayesian. However, some considerations are worth adding. Both the distance of a pair of LBP codes and the distance of a pair of BLOB codes are computed as the Hamming distance of a pair of binary codes of the same size. The codes in each pair have a similar expected amount of '0' and '1' due to the nature of the variations in the objects to be matched (irises). For this reason, the Hamming distances computed separately on each pair should not present scale variations due to the different coding methods, but rather due to an especially high difference among specific couples of irises. Moreover, both distances are computed by considering the same segmentation mask.

4. Experimental framework

The experiments to assess N-IRIS performances were mainly performed on the databases UBIRIS v1s2 (version 1 and session 2) (Proença and Alexandre, 2005) and UBIRIS v2 (version 2) (Proença et al., 2009), and following the protocols defined by the program committee of NICE II. In both cases RGB colour images are captured in visible light. UBIRIS v1 contains 1877 images with 800×600 pixels resolution, acquired from 241 subjects in two distinct sessions. UBIRIS v2 contains 11102 images of 400×300 pixels from both irises of 261 subjects. Tuning exploited 1000 images and corresponding segmentation masks provided to this aim by NICE II, together with a dedicated JAVA platform. N-IRIS was then tested by the NICE II evaluation commission on new images and masks, never provided before. The results were measured in terms of decidability, and through classical accuracy "figures of merit" Receiving Operating Curve (ROC), Equal Error Rate (EER), and Recognition Rate (RR). Decidability is defined as a function of mean and variance of intra- and inter-class scores. The higher the index, the better the discrimination ability of the system. If D^I and D^E denote the set of similarities resulting from intra- and inter-class matches, $\mu(D^I)$ and $\mu(D^E)$ the respective mean values, and $\sigma(D^I)$ and $\sigma(D^E)$ the standard deviations, the decidability index is:

$$d = \frac{\left|\mu(D^I) - \mu(D^E)\right|}{\sqrt{0.5 * \left(\sigma(D^I)^2 + \sigma(D^E)^2\right)}} \tag{6}$$

5. Results and discussion

All color images were converted in gray scale by assigning each pixel the weighted mean of the three primary channels of its RGB color. We tested LBP by dividing the images in horizontal or vertical bands and in blocks. LBP (n,m) will denote LBP execution on an image subdivided in n columns and m rows.

BLOB was run in single scale configuration, with fusion of different scale results, and with chaining (see Section 3.2). Scale t varied in the set $T = \{2,4,6,8,12,16,24\}$. In *fusion* and *chain* modes, we considered pairs (t_1,t_2) and triplets (t_1,t_2,t_3) of scales from T. BLOB

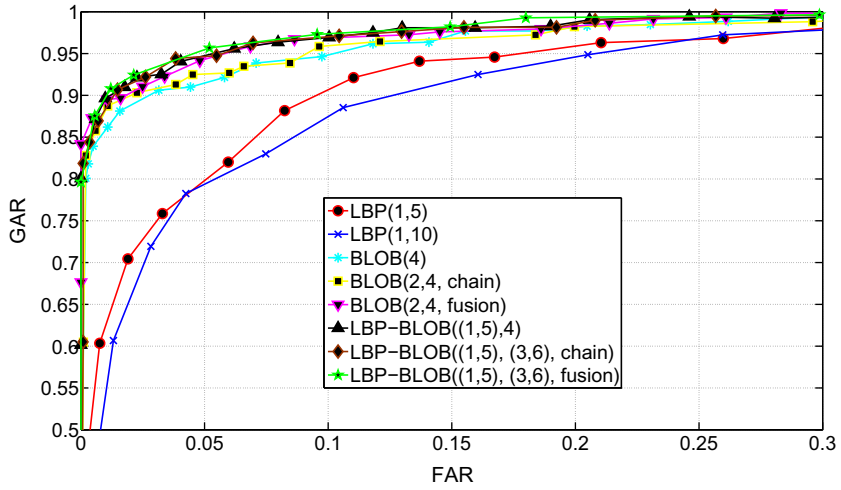

Fig. 7. Results from LBP, BLOB and LBP-BLOB with different configurations on UBIRIS v1 database.

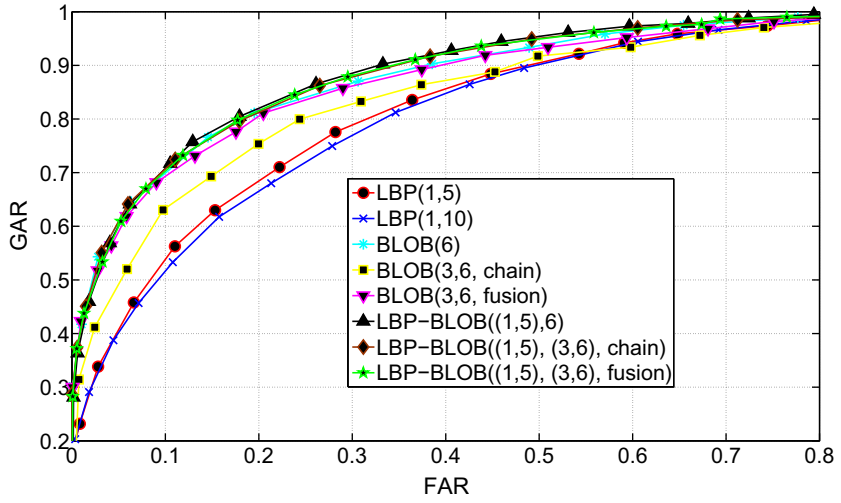

Fig. 8. Results from LBP, BLOB and LBP-BLOB with different configurations on UBIRIS v2 database.

(t_1) will denote single scale execution of BLOB at scale t_1, BLOB (t_1, t_2, *mode*) will denote the execution of BLOB in mode *mode* $\in$ {*chain*, *fusion*} for the pair of scales (t_1,t_2), and BLOB (t_1,t_2,t_3,*mode*) will denote the execution of BLOB in mode *mode* $\in$ {*chain*, *fusion*} for the triplet of scales (t_1,t_2,t_3). A configuration for LBP-BLOB combines single configurations for LBP and BLOB: LBP-BLOB (n,m,t_1), LBP-BLOB (n,m,t_1,t_2,*mode*) LBP-BLOB (n,m,t_1,t_2,t_3,*mode*).

Figs. 7 and 8 show the results from LBP and BLOB with different configurations, as well as different combinations of such configurations, on both UBIRIS v1 and UBIRIS v2. The subdivision in five horizontal bands seems an optimal LBP configuration for both databases. BLOB in *fusion* mode (the original one) provides better results than BLOB in *chain* mode. Moreover, BLOB works better with a single scale when using UBIRIS.v2. Though sounding strange, this is a consequence of the scarce clearness of most images in this database. For such images, using more scales provides poor benefit. On the contrary, the double scale in *fusion* mode seems the best BLOB configuration for UBIRIS.v1. BLOB seems to perform better than LBP, but this trend is reversed on low resolution images. This underlines a better ability by LBP to extract relevant features in these cases. We noticed that normalization fails in some critical situations, were the useful iris region is especially scarce and, at the same time, iris and pupil boundaries are not well separated as in the last row of Fig. 2. Matching problems encountered with LBP are related to excessive blurring, since the histogram undergoes a substantial alteration, while BLOB problems are related to irises with high off-axis angles which significantly alter blobs shape. Both Figs. 7 and 8 show that the LBP-BLOB performs better than the single methods. The combined method was tested within the provided JAVA framework on a dataset of 1000 images that the NICE II program committee extracted from UBIRIS v2, and its performances were measured in terms of decidability value. On such dataset, the method achieved a decidability value of 1.4825, while on the dataset used during the independent evaluation within NICE II, the obtained decidability value was 1.2565. We had no access to the latter dataset, so we cannot justify this difference.

6. Conclusions

This work presents an approach for matching irises captured in the visible light spectrum and in uncontrolled settings. The obtained images are subject to distortions such as occlusions from eyelids, reflexions or blurring. The approach exploits techniques for the local extraction of characteristic features from the iris

pattern. Linear Binary Patterns (LBP) and BLOB have been adapted and combined in an original and specific way, to address the difficult operational conditions due to the strongly relaxed capture constraints. The obtained results are quite satisfactory both in terms of ROC and of decidability value, most of all against the present research scenario, as the independent tests performed by NICE II program committee have demonstrated. This is a strong motivation to further improve performances. A very promising research line that we are following is the use of more local features, able to set off different iris peculiarities, as for example the directionality of extracted patterns. A second research line regards the strategy to combine results from different feature detectors and matchers, that might better exploit strengths and limits of each of them on images with different characteristics and capture settings.

Acknowledgments

The authors want to thank Emiliano Pigini and Davide Sammartino for the helpful proposals and implementation efforts.

References

Ahonen, T., Hadid, A., Pietikäinen, M., 2006. Face description with local binary patterns: Application to face recognition. IEEE Trans. PAMI 28, 2037–2041.

Chenhong, L., Zhaoyang, L., 2005. Efficient iris recognition by computing discriminable textons. In: International Conference on Neural Networks and Brain. pp. 1164–1167.

Chenhong, L., Zhaoyang, L., 2008. Local feature extraction for iris recognition with automatic scale selection. Image Vision Comput. 26, 935–940.

Daugman, J., 2004. How iris recognition works. IEEE Trans. Circuits Syst. Video Technol. 14, 21–30.

Mäenpää, T., Ojala, T., Pietikäinen, M., Soriano, M., 2000. Robust texture classification by subsets of local binary patterns. In: 15-th International Conference on Pattern Recognition. pp. 947–950.

Ojala, T., Pietikäinen, M., Mäenpää, T., 2002. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. PAMI 24, 971–987.

Pigini, E., 2010. Riconoscimento dell'Iride per Applicazioni di Domotica: Segmentazione e Normalizzazione. Master's thesis, Sapienza Univ. of Rome, Computer Science Dpt.

Proença, H., Alexandre, L., 2005. Ubiris: A noisy iris image database. In: 13th International Conference on Image Analysis and Processing - ICIAP 2005. Vol. LNCS 3617. pp. 970–977.

Proença, H., Filipe, S., Santos, R., Oliveira, J., Alexandre, L., 2009. The ubiris.v2: A database of visible wavelength images captured on-the-move and at-a-distance. IEEE Trans. PAMI 32 (8), 1529–1535.

Sammartino, D., 2010. Riconoscimento dell'Iride per Applicazioni di Domotica: Codifica e Matching. Master's thesis, Sapienza Univ. of Rome, Computer Science Dpt.

Sun, Z., Tan, T., Qiu, X., 2006. Graph matching iris image blocks with local binary pattern. In: International Conference on Biometrics. pp. 366–372.

Tan, T., Hea, Z., Sun, Z., 2010. Segmentation of visible wavelength iris images captured at-a-distance and on-the-move. Image Vision Comput. 28, 223–230.

Taubin, G., 1991. Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. IEEE Trans. PAMI 13, 115–1138.

Timo Ojala, M.P., Harwood, D., 1996. A comparative study of texture measures with classification based on featured distributions. Pattern Recognit. 29, 51–59.