Efficient Iris Spoof Detection via Boosted Local Binary Patterns

Zhaofeng He, Zhenan Sun, Tieniu Tan, and Zhuoshi Wei

Center for Biometrics and Security Research
National Laboratory of Pattern Recognition, Institute of Automation
Chinese Academy of Sciences, P.O. Box 2728, Beijing, P.R. China, 100190
{zfhe,znsun,tnt,zswei}@nlpr.ia.ac.cn

Abstract. Recently, spoof detection has become an important and challenging topic in iris recognition. Based on the textural differences between the counterfeit iris images and the live iris images, we propose an efficient method to tackle this problem. Firstly, the normalized iris image is divided into sub-regions according to the properties of iris textures. Local binary patterns (LBP) are then adopted for texture representation of each sub-region. Finally, Adaboost learning is performed to select the most discriminative LBP features for spoof detection. In particular, a kernel density estimation scheme is proposed to complement the insufficiency of counterfeit iris images during Adaboost training. The comparison experiments indicate that the proposed method outperforms state-of-the-art methods in both accuracy and speed.

1 Introduction

With the increasing demands of security in our daily life, iris recognition has rapidly become a hot research topic for its potential values in personal identification [1,2,3,4]. As shown in Fig. 1(a) and (b), the iris of a human eye is the annular part between the black pupil and white sclera. It displays rich texture that is commonly thought to be highly discriminative between eyes and stable over individuals' lifetime, which makes iris particularly useful for personal identification.

However, it must be aware that, as any other authentication technique, iris recognition is also possibly forged and illegally used [4]. Several potential iris counterfeits have already been considered, e.g., printed iris, re-played video, fake glass/plastic eye, printed contact lens, etc. Among these counterfeits, printed contact lens is commonly thought to be particularly dangerous [5,6]. For the sake of convenience, more and more people wear contact lens. Once they enrolled into an iris system without taking off their contact lens, a big concern rises: anyone who wears the same contact lens can be possibly mistaken as the authorized user even the imposter does not intend to (whereas other counterfeits, e.g. the printed iris, usually take an illegal initiative). Figure 1(c) shows one example contact lens and the resultant iris images by different eyes wearing it. We can see that although the individual live iris textures are quite different from each other, the contact lens wearing iris images look almost the same. It has been reported by several researchers [7,8] that "it is actually possible to spoof some iris recognition systems with

M. Tistarelli and M.S. Nixon (Eds.): ICB 2009, LNCS 5558, pp. 1080–1090, 2009.

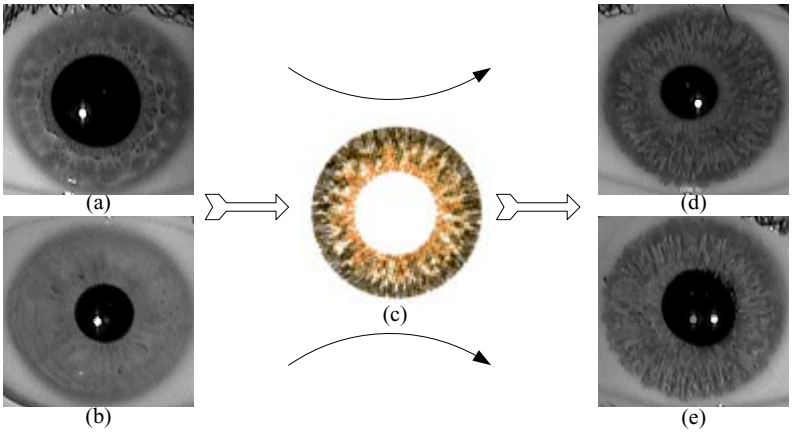

Fig. 1. Example of contact lens and the resultant iris images on different eyes. (a)-(b): The live iris images from two different eyes. (c) The contact lens. (d)-(e): The resultant iris images of (a) and (b) after wearing (c). Although the genuine live iris textures of the two eyes are different, their contact lens wearing iris images seem almost the same.

well-made contact lens" [5]. It is therefore important and desirable to detect the counterfeit iris (especially the contact lens) before recognition to avoid unnecessary losses.

A few researchers have contributed several methods for iris spoof detection. Some of them suggested spoof detection via turning on/off the illuminators followed by checking the responses on the resultant images (e.g., the pupil hippus [4], the specular spots on the cornea [8], and so on). For example, Lee et al. [8] proposed a fake iris detection scheme via investigating the specular spots of collimated IR-LED. Such illuminator based methods can be useful for printed iris or glass/plastic eyes, but tend to fail for contact lens. Moreover, they require additional hardware and have to capture a series of iris images for analysis, which inevitably increase the hardware cost and the recognition time.

Several software-based methods have also been proposed. Daugman [9] and Tan et al. [4] suggested detecting the printed iris via frequency analysis. The basic idea is to utilize the frequency characteristics of the printed iris due to the periodic dot printing. Obviously, this method is limited to printed iris detection. Recently, He et al. [5] proposed a contact lens detection method via statistical texture analysis. Four distinctive features based on gray level co-occurrence matrix (GLCM) are extracted. Support vector machine is used for classification. In [6], Wei et al. also proposed a texture analysis based scheme for contact lens detection. In their work, Iris-Textons are learned and used for texture representation. Both of the texture based methods achieved encouraging performance.

From the above description, we can conclude that the hardware-assisted methods utilize the physical characteristics of the pupil and eye, and are effective for printed iris, replayed video or glass eyes. While the texture based methods focus on textural differences between the live and counterfeit iris images, and are effective for contact lens detection. In this paper, we propose a more efficient texture based method for iris spoof detection (especially the contact lens detection). Firstly, we divide the valid part

of the iris into sub-regions according to the properties of iris textures. Local binary patterns (LBP) are then adopted for representing the statistical texture characteristics of each sub-region. Finally, the well-known Adaboost learning algorithm is performed to select the most discriminative LBP features for spoof detection.

The rest of the paper is organized as follows: in Section 2, we describe how the iris image is preprocessed for spoof detection purpose according to the properties of iris textures. In Section 3, the local binary patterns are adopted for effective representation of the iris texture. In Section 4, Adaboost learning is performed to select the most discriminative LBP features for spoof detection. In particular, we present a novel kernel density estimation scheme to tackle the lack of sufficient training counterfeit iris images. The experiments and discussions are presented in Section 5 prior to the conclusions in Section 6.

2 Iris Properties and Preprocessing

Although iris textures are commonly thought to be highly discriminative between eyes, they (including the contact lens wearing iris textures) still present several desirable common properties [10], such as:

1) *The radial distribution:* Even within an iris, the scale of the iris micro-structures varies a lot along the radius. Usually the larger the radius is, the bigger the iris micro-structures will be (see Fig. 2(a)).
2) *The angular self-similarity:* Although different angular regions remain discriminative, their texture patterns display a certain degree of consistence/correlation as shown in Fig. 2(b).

These properties suggest dividing the iris into multiple regions as shown in Fig. 2(c). We can see that each sub-region contains a particular texture pattern. Via such division, more specific representation of iris can be obtained, and hence makes it easier to discriminate live iris textures from counterfeit iris textures.

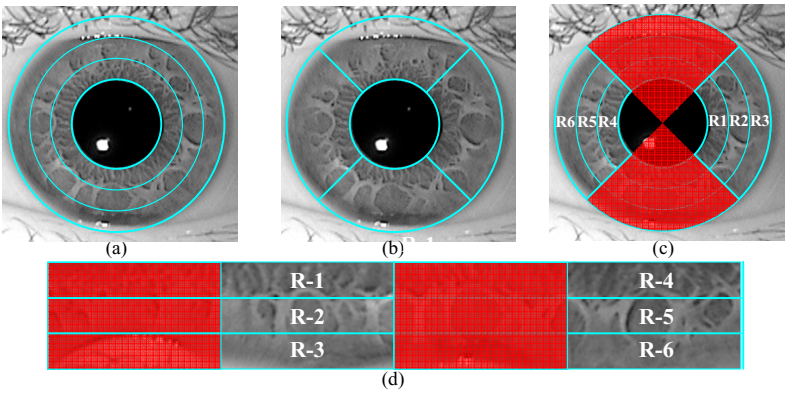

Fig. 2. The properties of iris textures and preprocessing of the iris image. After excluding the upper and lower quarters, the ROI of the iris is divided into six sub-regions.

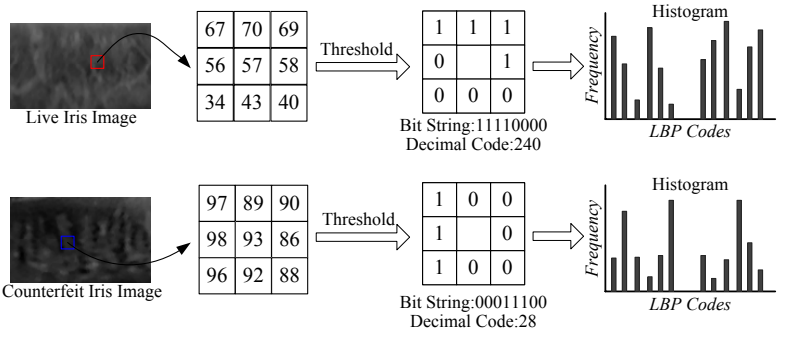

Fig. 3. Local binary patterns encoding. The LBP histograms of different textures are different, and therefore can be used as a texture descriptor.

Moreover, the upper and lower parts of the iris are almost always occluded by eyelids or eyelashes [11]. It is therefore straightforward to exclude the upper and the lower quarters from feature extraction for a more concise representation. And in order to achieve translation and scale invariance, the iris is normalized to a rectangular block of a fixed size of 64*512 [4]. The image preprocessing scheme is illustrated in Fig. 2. As can be seen, the whole iris image is divided into three annular sections along the radial direction, and two sectors along the angular direction. In total, we get six sub-regions.

3 Local Binary Pattern Encoding

As mentioned in Section 1, the textures of counterfeit iris images and live iris images are different in appearance and can be used for spoof detection. But how to encode such differences remains an open problem. Recently, the local binary patterns (LBP) [12] has emerged as an effective texture descriptor. Basically, LBP is defined for each pixel by thresholding its 3*3 neighborhood pixels with the center pixel value, and considering the result as a binary bit string, see Fig. 3. Each LBP code represents a type of micro image structure, and the distribution of them can be used as a texture descriptor [12]. The original LBP is later extended to multi-scale LBP (denoted by $LBP_{P,R}$) and uniform LBP (denoted by LBP^{u2}). $LBP_{P,R}$ is calculated by thresholding P equally spaced points on a circle (whose radius is R) with the center pixel value. A LBP code is called uniform if its bit string contains at most two bit-wise transitions from 0 to 1 or vice versa. LBP based methods have been proved to be successful in biometric texture representation, such as face [13] and iris [14].

In this work, we adopt multi-resolution LBPs (namely, $LBP^{u2}_{8,1}$, $LBP^{u2}_{8,2}$, $LBP^{u2}_{8,5}$, $LBP^{u2}_{8,7}$, $LBP^{u2}_{12,2}$, $LBP^{u2}_{12,3}$, $LBP^{u2}_{12,5}$, $LBP^{u2}_{16,3}$, $LBP^{u2}_{16,5}$, $LBP^{u2}_{16,7}$) for texture representation of each sub-region obtained in Section 2. The number of bins of them is (59, 59, 59, 59, 135, 135, 135, 243, 234, 234) respectively. As described in Section 2, we have 6 sub-regions, hence we will totally get (59+59+59+59+135+135+135+243+243+243) $*$ 6=8220 possible LBP bins. Each bin represents the frequency of one type of micro image structures on one sub-region, and is considered as a candidate texture feature. A large pool of regional LBP features (LBP bins) is therefore generated. Definitely, this

feature pool must contain much redundant information because of the redundancy between different LBP features as well as that between different sub-regions. To learn the most discriminative regional LBP features from the redundant feature pool, we turned to the following Adaboost algorithm.

4 Adaboost Learning

Adaboost is a well-known machine learning algorithm that can select a small set of the most discriminative features from a candidate feature pool [10,15]. It is particularly efficient for binary (two-class) problems, and therefore is suitable for selecting the best LBP features for iris spoof detection.

4.1 Adaboost Learning

Given that $\{x_i, y_i\}_{i=1}^{N}, (x \in R^d, y \in \{+1, -1\})$ is N labeled training samples with associated weights $\{w(x_i)\}_{i=1}^{N}$; $\Phi = \{\phi_m(\cdot): R^d \rightarrow R\}_{m=1}^{M}$ is a candidate feature pool of x; our goal is to automatically learn a small set of the most discriminative features $\{\phi_t\}_{t=1}^{T}$ from the feature pool, and construct an ensemble classifier:

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} h_t(\phi_t(x))\right) \tag{1}$$

where $h_t(s) : R \rightarrow R$ is a component classifier that can output a 'confidence' of x being a positive when $\phi_t(x)$ equals s [10]. In this work, x is an iris image and $\{\phi_m(x)\}_{m=1}^{M}$ corresponds to the LBP features (i.e., LBP bins).

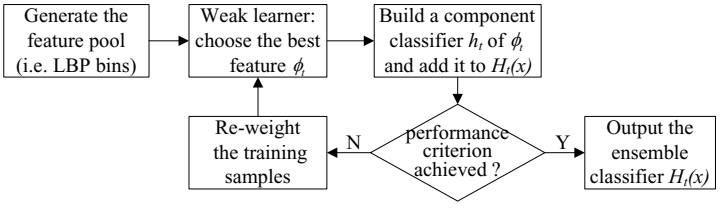

Fig. 4. A generalized framework of Adaboost Learning

The flowchart of Adaboost learning is depicted in Fig. 4 [10]. It begins with generating the feature pool on the training samples. After that, Adaboost repeatedly learns the component classifiers $h_t(\phi_t(\cdot))$ on the weighted versions of the training samples until the performance criterion is satisfied. Clearly, there are three key modules involved in Adaboost: the weak learner, the component classifier and the re-weighting function.

The weak learner is essentially the criterion for choosing the best feature (e.g., $\phi_t(\cdot)$) on the weighted training set. The component classifier h_t outputs a confidence score of x being a positive based on its ϕ_t value. The re-weighting function maintains a distribution over the training samples and updates it in such a way that the subsequent component

classifier can concentrate on the hard samples by giving higher weights to the samples that are wrongly classified by previous classifier.

Among various Adaboost algorithms, we choose the confidence-rated Adaboost learning [15] for its efficiency and simplicity. In confidence-rated Adaboost, the weak learner tries to find the feature that can maximizes the following criterion:

$$\phi_t = \arg\min_{\phi \in \Phi} 2 \sum_{j=1}^{N} \sqrt{P_w^+(\phi^j(x))\, P_w^-(\phi^j(x))} \tag{2}$$

where $P_w^+(\phi_m(x))$, $P_w^-(\phi_m(x))$ are the positive and negative probability distributions of $\phi_m(x)$ on the weighted training set (see Fig. 5). The corresponding component classifier is constructed as follows:

$$h_t(\phi_t) = \frac{1}{2} \ln \frac{P_w^+(\phi_t)}{P_w^-(\phi_t)} \tag{3}$$

The re-weighting function is as follows.

$$w_{t+1}(x_i) \leftarrow w_t(x_i) \exp(-y_i h_t(\phi_t(x_i))) \tag{4}$$

Please refer to [10] and [15] for more details of Adaboost learning.

4.2 Kernel Density Estimation for Counterfeit Iris Images

From Eq. 2 and Eq. 3 we can see that both the weak learner and the component classifier are dependent on the density distributions of positive and negative samples, which are estimated by histograms. More bins in the histogram give a more refined representation of the feature density distribution. However, when the training samples of one class are insufficient (e.g., due to the difficulty of collection), samples dropped into each bin is not enough for a stable estimation of the distribution of this class but just an ad-hoc one of the current training samples. As a result, the classifier learned based on the limited training samples will be ad-hoc, i.e., has low generalization capability and is sensitive to possible noise (see Eq. 3).

A possible solution to this problem is the kernel density estimation (KDE) [16]. The basic idea of kernel density estimation is that: if the feature value of one training sample is x, it is highly possible that there exist several similar samples whose feature values are around x. Suppose $p(x)$ is the probability density of a LBP feature, and we wish to estimate $p(x)$ via a random sample set $x_1, x_2, \ldots, x_N$. In KDE, $p(x)$ is estimated as follows:

$$\hat{p}(x) = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{h} k\left(\frac{x - x_n}{h}\right) \tag{5}$$

where $k\left(\frac{x-x_n}{h}\right)$ is a kernel of width h. From Eq. 5, we can see that the density on x is interpreted as the sum of N local kernels centered on the N data points x_n.

For the sake of smoothness, a popular choice for $k\left(\frac{x-x_n}{h}\right)$ is the Gaussian, which gives rise to the following kernel density model [16]:

$$\hat{p}(x) = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{(2\pi h^2)^{1/2}} exp\left\{-\frac{|x - x_n|^2}{2h^2}\right\} \tag{6}$$

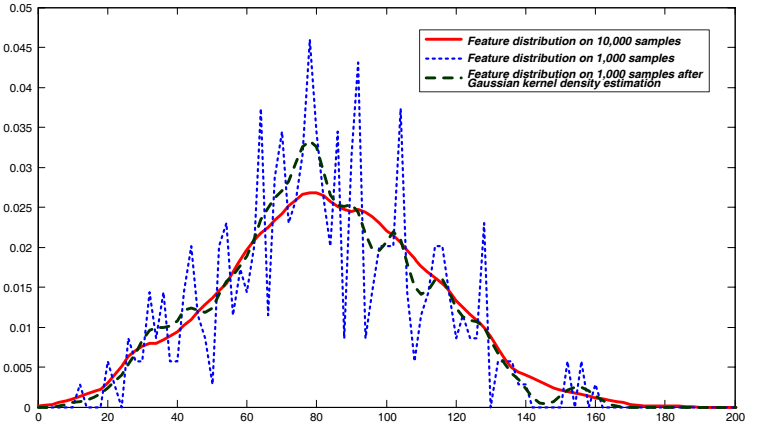

Fig. 5. The distributions of $LBP^{u2}_{8,2}('1001011')$ on 10, 000 iris images, 1, 000 iris images, and the Gaussian kernel density estimation (KDE) version of the 1, 000 one respectively. We can see that although the 1, 000 distribution is quite different to the 10, 000 one, the Gaussian KDE one looks much similar with the 10, 000 one.

where h denotes the standard deviation of the Gaussian kernel. Thus our density model is obtained by placing a Gaussian over each data point and then adding up the contributions over the whole data set. Clearly, the contribution (weight) of x_n decreases while its distance from x increasing.

The usefulness of Gaussian kernel density estimation is illustrated in Fig. 5, where the distribution obtained on 10,000 iris images is surprisingly similar with the distribution obtained on only 1,000 iris images with Gaussian kernel density estimation. This indicates that: although we cannot obtain the genuine distribution of 10, 000 iris images (due to the difficulty of collection), at least we can estimate a much closer one to it via Gaussian kernel density estimation, which, as demonstrated in our experiments, efficiently complements the lack of sufficient training counterfeit iris samples. A notable point in Gaussian KDE is the setting of h which controls the trade-off between having $\hat{p}(x)$ close to the data (at small h) and having $\hat{p}(x)$ smooth (at large h). Experimental results show that $h = 1.8$ is a good choice for this work.

5 Experimental Results

5.1 The Data Set

Experiments are performed to evaluate the usefulness of the proposed method. Due to the absence of public counterfeit iris image database, we manually collected 600 counterfeit iris images. While a minority of these iris images come from printed iris and glass eye, the majority of them are printed color contact lens iris images since this work focuses on contact lens detection. In detail, this counterfeit iris database consists of 20 kinds of different contact lens, with varying textures printed onto them. Some of the counterfeit iris images are shown in Fig. 6. We can see that some of them are difficult even for human to decide.

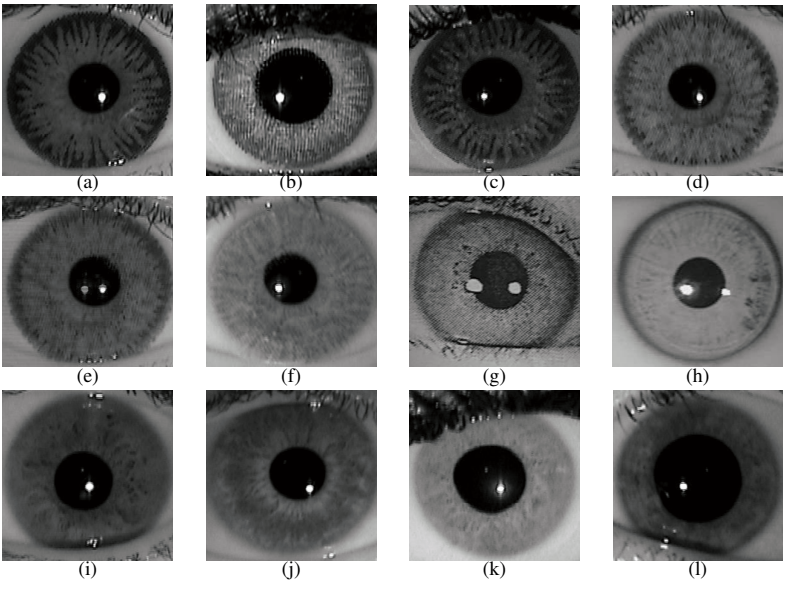

Fig. 6. Examples of training samples. (a)-(f): Contact lens wearing iris images. (g) Printed iris. (h) Glass eye. (i)-(l): Live iris images. We can see that some of the samples are difficult even for human to decide.

Although there are not sufficient counterfeit iris images, fortunately, we can collect sufficient live iris images. This partly complements the potential performance degradation due to the absence of sufficient counterfeit iris images. The live iris images are randomly selected from two well-known iris image databases, namely CASIA-Iris-V3 [17] and the ICE v1.0 [18]. In total, about 10, 000 live iris images are collected, which can cover almost all kinds of textures of the live irises.

5.2 Adaboost Learning

300 counterfeit iris images and 6000 live iris images are randomly selected from the above data set for Adaboost training. As described in Section 3, 8220 LBP features (i.e., LBP bins) are generated for Adaboost learning (during which Gaussian kernel density estimation is applied to complement the insufficiency of counterfeit iris images). The learned Adaboost classifier contains only 85 features, i.e., 85 LBP bins. The first twelve selected LBP features are shown in Fig. 7. We can see that the selected LBP features are in different sub-regions and different scales, which indicates the successfulness of iris division and multi-resolution LBP analysis. Moreover, many different LBP patterns are learned to represent the rich textures of the live and counterfeit iris images. We achieve 100% correction on the training set thanks to the aggressive learning ability of Adaboost [10]. But a big concern is its generalization capability on the test set.

5.3 Test Results and Discussions

The learned Adaboost classifier is tested on the remaining iris images. It is interesting to compare our method with the methods of He [5] and Wei [6]. Table 1 shows the

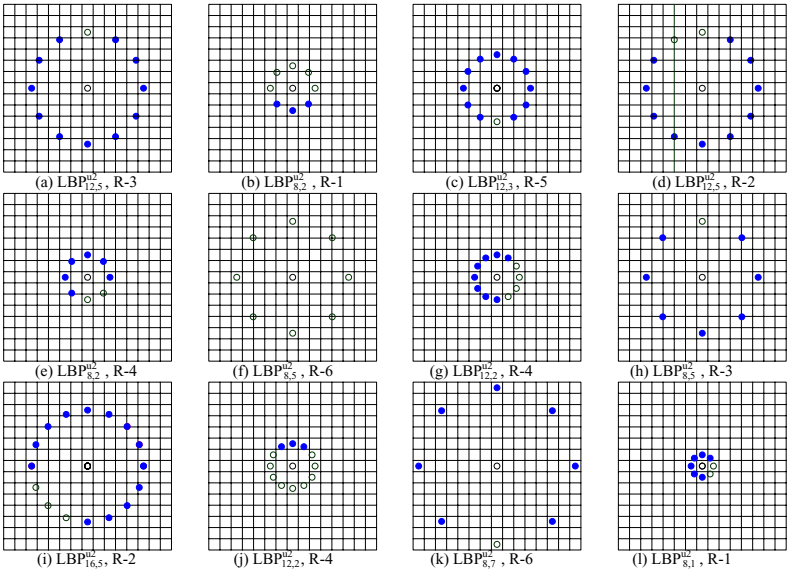

Fig. 7. The first twelve LBP features learned by Adaboost. Note that the learned LBP patterns are in different sub-regions and different scales.

experimental results, where FAR (False Accept Rate), FRR (False Reject Rate) and speed are presented. We can see that the proposed method outperforms the other methods both in accuracy and speed. The encouraging accuracy is perhaps due to four reasons:

1. The local binary patterns are effective in representing the iris textures.
2. The division of the iris into sub-regions enables a more specific representation of the iris textures.
3. The Adaboost learning is efficient in learning the most discriminative features for spoof detection.
4. The proposed Gaussian kernel density estimation scheme partly complements the insufficiency of counterfeit iris images, and increases the generalization capability of the learned Adaboost classifier.

Clearly, the fast execution is due to the computational simplicity of the LBP compared with the calculation of GLCM [5] or iris-textons [6].

Table 1. Overall performance of the learned classifiers via [5], [6] and the proposed method

Algorithm	FAR (%)	FRR (%)	Speed(ms)
He [5]	4.33	6.84	230
Wei [6]	3.67	6.91	340
Proposed	0.67	2.64	160

6 Conclusions

In this paper, we propose a texture analysis based method for efficient iris spoof detection (especially for contact lens detection). The basic idea is the textural differences between counterfeit iris images and the live iris images. Local binary patterns are adopted for representing the textural characteristics of local sub-regions, and Adaboost learning (together with Gaussian kernel density estimation) is performed to select the most discriminative LBP features for spoof detection. Extensive experiments indicate that the proposed method can be well adapted for iris spoof detection.

Acknowledgement

This work is supported by research grants from the National Basic Research Program (Grant No. 2004CB318110), the Natural Science Foundation of China (Grant No. 60723005, 60736018, 60702024), NLPR 2008NLPRZY-2, the National Hi-Tech Research and Development Program of China (2006AA01Z193, 2007AA01Z162).

References

1. Jain, A.K., Ross, A., Prabhaker, S.: Jain, Arun Ross, and Salil Prabhaker. An introduction to biometric recognition. IEEE Trans. on Circuits and Systems for Video Technology 14(1), 4–20 (2004)
2. Daugman, J.: How iris recognition works. IEEE Trans. On Circuits and Systems for Video Technology 14(1), 21–30 (2004)
3. Wildes, R.: Iris recognition: An emerging biometric technology. Proceedings of the IEEE 85, 1348–1363 (1997)
4. Ma, L., Tan, T., Wang, Y., Zhang, D.: Personal identification based on iris texture analysis. IEEE Trans. on Pattern Analysis and Machine Intelligence 25(12), 1519–1533 (2003)
5. He, X., An, S., Shi, P.: Statistical texture analysis based approach for fake iris detection using support vector machine. In: Proc. of Int'l Conf. on Biometrics 2007, pp. 540–546 (2007)
6. Wei, Z., Qiu, X., Sun, Z., Tan, T.: Counterfeit iris detection based on texture analysis. In: Proc. of IEEE Int'l Conf. on Pattern Recognition (ICPR 2008) (2008)
7. Daugman, J.: Iris recognition and anti-spoof countermeasures. In: Proc. of the 7th Int'l Biometrics Conference (2004)
8. Lee, E.C., Park, K.R., Kim, J.: Fake iris detection by using purkinje image. In: Zhang, D., Jain, A.K. (eds.) ICB 2005. LNCS, vol. 3832, pp. 397–403. Springer, Heidelberg (2005)
9. Daugman, J.: Demodulation by complex-valued wavelets for stochastic pattern recognition. Intl. Journal of Wavelets, Multi-resolution and Information Processing 1, 1–17 (2003)
10. He, Z., Tan, T., Sun, Z., Qiu, X.C.: Boosting ordinal features for accurate and fast iris recognition. In: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR 2008) (2008)
11. He, Z., Tan, T., Sun, Z., Qiu, X.C.: Towards accurate and fast iris segmentation for iris biometrics. IEEE Trans. on Pattern Analysis and Machine Intelligence (accepted, 2008)
12. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. on Pattern Analysis and Machine Intelligence 24(7), 971–987 (2002)

13. Ahonen, T., Hadid, A., Pietikäinen, M.: Face description with local binary patterns: Application to face recognition. IEEE Trans. on Pattern Analysis and Machine Intelligence 28(12), 2037–2041 (2006)
14. Sun, Z., Tan, T., Qiu, X.: Graph matching iris image blocks with local binary pattern. In: Proc. of 1st Int'l Conf. on Biometrics, Hong Kong, pp. 366–372 (2006)
15. Schapire, R.E., Singer, Y.: Improved boosting algorithms using confidence-rated predictions. Machine Learning 37, 297–336 (1999)
16. Bishop, C.M.: Pattern Recognition and Machine Learning. In: Information Science and Statistics, ch. 6, pp. 291–324. Springer, New York (2006)
17. Chinese Academy of Sciences Institute of Automation. CASIA Iris Image Database Version 3.0, http://www.cbsr.ia.ac.cn/IrisDatabase.htm
18. Ice v1.0 iris image database, iris challenge evaluation (ice), http://iris.nist.gov/ice/