Pattern Recognition ■ (■■■■) ■■■–■■■

Contents lists available at ScienceDirect

Pattern Recognition

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

Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients

Fengyi Song, Xiaoyang Tan*, Xue Liu, Songcan Chen

Department of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Yudao Street 29, Nanjing 210016, China

ARTICLE INFO

Article history:
Received 25 June 2013
Received in revised form
17 February 2014
Accepted 23 March 2014

Keywords:
Eye closeness detection
Eye state measurement

ABSTRACT

In this paper, we present a novel approach to deal with the problem of detecting whether the eyes in a given still face image are closed, which has wide potential applications in human–computer interface design, facial expression recognition, driver fatigue detection, and so on. The approach combines the strength of multiple feature sets to characterize the rich information of eye patches (concerning both local/global shapes and local textures) and to construct the eye state model. To further improve the model's robustness against image noise and scale changes, we propose a new feature descriptor named Multi-scale Histograms of Principal Oriented Gradients (MultiHPOG). The resulting eye closeness detector handles a much wider range of eye appearance caused by expression, lighting, individual identity, and image noise than prior ones. We test our method on real-world eye datasets including the ZJU dataset and a new Closed Eyes in the Wild (CEW) dataset with promising results. In addition, several crucial design considerations that may have significant influence on the performance of a practical eye closeness detection system, including geometric normalization, feature extraction, and classification strategies, are also studied experimentally in this work.

1. Introduction

As one of the most salient facial features, eyes, which reflect the individual's affective states and focus attention, have become one of the most important information sources for face analysis. Efficiently and accurately understanding the states of the eyes in a given face image is therefore essential to a wide range of face-related research efforts, such as human–computer interface design, facial expression analysis, driver fatigue monitoring, liveness detection.

The task of eye closeness detection is to decide whether the eyes are closed.[1] This task is challenging since the degree of eye closeness may be different for each face and there are many ambient factors that may significantly change the appearance of the eyes, such as lighting, pose, scales and imaging conditions [2] (as shown in Fig. 1). In addition, inaccurate eye localization may introduce a great difficulty to this problem.

Previously, eye closeness was commonly served as the trigger of a series of affection states and also physiology states in many applications [4–11]. However, many of these works did not focus on the task of eye closeness detection but simply treated it as a preprocessing step in the overall pipeline of a particular application. In these methods (we call them feature-based methods in the remaining text), typical geometrical characteristics, such as visible iris and elliptical shape of eyelids [10–16,5,4,7], are extracted as the evidence distinguishing closed eyes from open ones. Other commonly used features are distinct intensity-distribution patterns between open eyes and closed eyes caused by the presence/absence of iris and eye white. For example, by accumulating gray intensity along the horizontal or vertical direction on the coarse eye region, the resulted projection curves show different shapes between the closed eyes and the open ones [17–19,8,20]. Such curves actually reflect the global intensity distribution and are vulnerable to inaccurate eye region localization and various ambient environment changes.

Appearance-based methods for eye states detection have attracted much attention recently, which try to extract useful visual features from the photometric appearance of the eyes. One major advantage of this type of methods compared to feature-based ones is that they could provide richer and more reliable information for the subsequent classification, particularly so for low-quality images. In particular, these methods first extract

* Corresponding author. Tel.: +86 25 8489 2956; fax: +86 25 8489 2452.
E-mail addresses: f.song@nuaa.edu.cn (F. Song), x.tan@nuaa.edu.cn (X. Tan), liuxue@nuaa.edu.cn (X. Liu), s.chen@nuaa.edu.cn (S. Chen).

[1] Note that in this paper the term "eye closeness" should not be confused with "interocular distance", and its meaning is different from "eye blink" as well – the latter usually refers to a short sequence of eye states changes (including eye closeness) [1] and closeness is just one possible state of eyes among others.

http://dx.doi.org/10.1016/j.patcog.2014.03.024
0031-3203/

Fig. 1. Illustration of some commonly encountered challenges in the task of eye closeness recognition in real-scenarios with variations caused by individuals, lighting, blur, occlusion, and disguise. The first row shows face images in our CEW dataset with eyes closed, while the second row shows face images from the LFW dataset [3] with both eyes open.

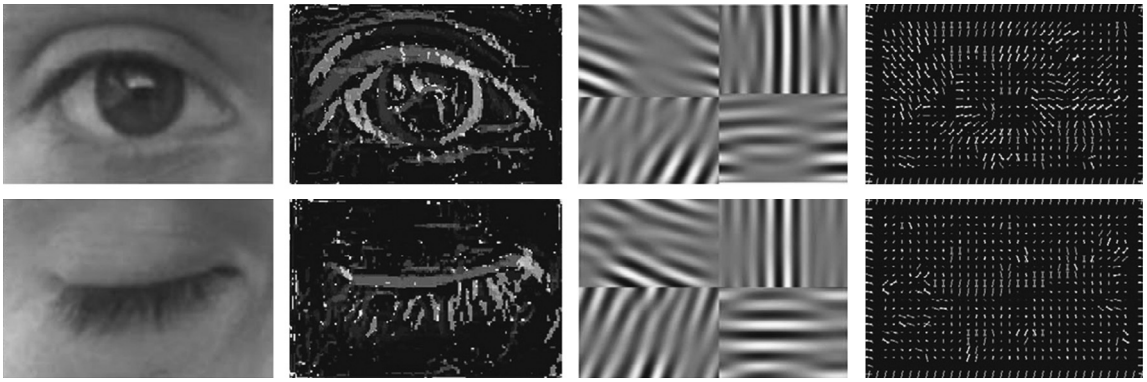

Fig. 2. Visualization of three kinds of descriptors for an open eye (top) and a closed eye (bottom). The original eye images are shown in the leftmost column and their three feature descriptors are listed from the left to the right: Local Ternary Patterns (LTP) [30], Gabor wavelet [31] and Histograms of Principal Oriented Gradients (HPOG) (proposed in this paper).

various kinds of mid-level features depicting basic shape and texture information of the target images (see Fig. 2), such as Local Binary Patterns (LBP) [21–23], Gabor wavelets [24–27], and then automatically analyze discriminant patterns by resorting to advanced machine learning tools, such as Adaboost [28,29,23], Support Vector Machine (SVM) [21,25,22,26] and neural network [24]. Consequently, appearance-based methods behave robustly even under very challenging imaging conditions.

While the aforementioned methods are very efficient and successful in dealing with input images that are in relatively good condition ensuring that the needed features are captured clearly, they may be vulnerable to inaccurate eye position estimation, illumination changes, image blur, and eye occlusion. More robust methods are hence desired, especially under the challenging real-life application scenarios. To meet these requirements, we have acquired a new eye state dataset called Closed Eyes in the Wild (CEW) from the Internet. Contrary to datasets that are collected systematically in the laboratory, very few preconditions are set in our dataset except that the faces with close eyes in the acquired images are detectable by the state-of-the-art Voila-Jones face detector [32] and the eyes in the face are localizable by our eye localizer [33]. As illustrated in Figs. 1 and 8, the eyes in our dataset contain much wilder variations than previous datasets such as the BoiID [34], AR [35], and CAS-PEAL [36]. Regarding the size of our dataset, there are over 1000 faces with the eyes closed available. More details about our dataset will be given in Section 3.

To address the problem of eye closeness detection under uncontrolled conditions, this paper intends to integrate various feature descriptors for robust and discriminant representation. In particular, we combine the strength of multiple feature sets to characterize local/global shapes and local textures of eye patches, and construct the eye state model based on such representation. To further improve the model's robustness against image noise and scale changes, we propose a new feature descriptor named Multi-scale Histograms of Principal Oriented Gradients (Multi-HPOG). Extensive experimental results on real-world eye datasets including the ZJU dataset and a new Closed Eyes in the Wild (CEW) dataset show that the proposed approach handles a much wider range of eye appearance caused by expression, lighting, individual identity, and image noise than prior ones. In addition, several crucial design considerations that may have significant influence on the performance of a practical eye closeness detection system, including geometric normalization, feature extraction, and classification strategies, are identified and investigated experimentally in this work. A preliminary version of this work appeared in [37].

In what follows, we present the architecture of our eye closeness detection system in Section 2. Under this architecture, we describe the feature descriptors used in this work besides the newly proposed one in Section 2.2 and discuss our classifiers in Section 2.3. Extensive experimental results are given in Section 3. Finally, we conclude this work in Section 4.

Please cite this article as: F. Song, et al., Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.024

2. Overall architecture of the proposed system

The overall architecture of the proposed eye closeness detection system is given in Fig. 3. For a given test image, we detect and crop the face portion by a Haar-like feature based ensemble detector used by Viola and Jones [32], then adopt the discriminative pictorial structural model [33] to localize eyes. With the localized position of the eye, we further refine the eye region and align it by an information-theory geometric normalization method used in [38]. On the aligned eye patch, various feature sets are extracted and are input into the classifier for the final decision.

2.1. Eye patch alignment

One of the key components of our system lies in the inclusion of the eye patch alignment module. This is based on the following considerations: (1) eyes in a face image may undergo various in-plane/out-of-plane pose changes or scale changes, and most feature sets (e.g., Histograms of Oriented Gradients (HOG) [39], Local Ternary Patterns (LTP) [30], and Gabor wavelets [31]) are not fully invariant to such variations; (2) even we have discarded those patches with very low positive responses from further analysis (about 0.5% of all images), our automatic eye localizer may not perform so accurate in any case. Therefore, performing geometric alignment is necessary as a preprocessing step to improve robustness against pose and scale changes and against inaccurate eye localization.

However, one difficulty of this approach is that it is hard to find anchor points for eye patches and hence the traditional anchor-points-based alignment method cannot be applied. Here, we adopt an information-theory geometric normalization method originally proposed for medical image registration, i.e., the congealing method [38]. This is an unsupervised image normalization method which learns a particular affine transformation for each eye patch such that the entropy of a group of them is minimized. Fig. 17 gives some illustrations of eye patches normalized using this method, from which we can see that the locations of the eyes are centered, and their sizes are scaled.

2.2. Feature sets

In this work, we consider three types of feature descriptors, which can capture rich image information such as the local shape, the global shape, and the local texture even under difficult conditions. In particular, we use a variant of Histograms of Oriented Gradients (HOG) [39], Local Ternary Patterns (LTP) [30], and Gabor wavelets [31] feature sets for our purpose.

2.2.1. Local shape descriptor

One typical local shape descriptor is the Histogram of Oriented Gradients (HOG) proposed by Dalal and Triggs [39], which has proved a very successful feature descriptor in computer vision. The key idea of this descriptor is to pool the local orientation (shape) information instead of the magnitude of small image patches. In particular, an image is divided into sets of small spatial regions called "*cells*", and several neighboring cells constitute a larger local region called "*block*", which is the basic component of the descriptor. The local shape information is first extracted on every pixel of a cell by calculating its gradient, and is pooled in that cell as well as in other cells within the same block (but with different weights according to the spatial distance), which helps to improve the "smoothness" of the resulting distribution representation (a histogram actually). The final histogram of each block is undergone a contrast-normalization before being concatenated to form the final descriptor. This makes the descriptor robust against small changes in illumination or shadowing.

In this work, we present two variants of the HOG descriptor, named Histograms of Principal Oriented Gradients (HPOG) and multi-scale HPOG (MultiHPOG). Fig. 4 gives the flow chart of these descriptors, which will be detailed in what follows.

Histograms of Principal Oriented Gradients (*HPOG*): Note that HOG does not consider the unstableness of the computed gradients. This is because pixel-wise gradients are sensitive to appearance changes caused by image blur, noise, low resolution, and so on. To address this problem, instead of using the pixel-based gradient computation directly, we consider the possibility to do this in a larger scope and propose the Histograms of Principal Oriented Gradients (HPOG) descriptor.

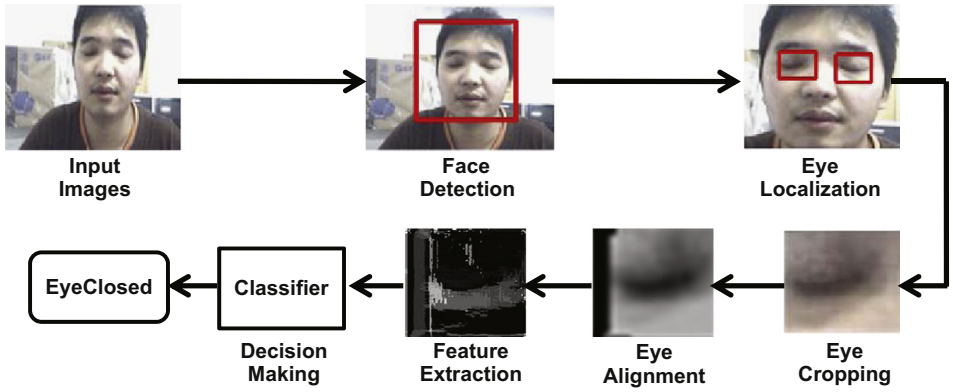

Fig. 3. The overall architecture of the proposed eye closeness detection system.

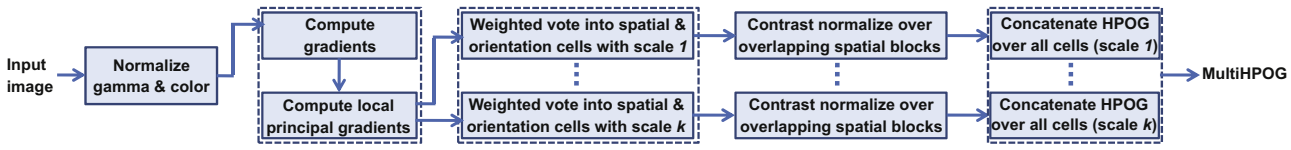

Fig. 4. The flow chart of the proposed Histograms of Principal Oriented Gradients (HPOG) descriptor and its multi-scale variant (MultiHPOG).

Please cite this article as: F. Song, et al., Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.024

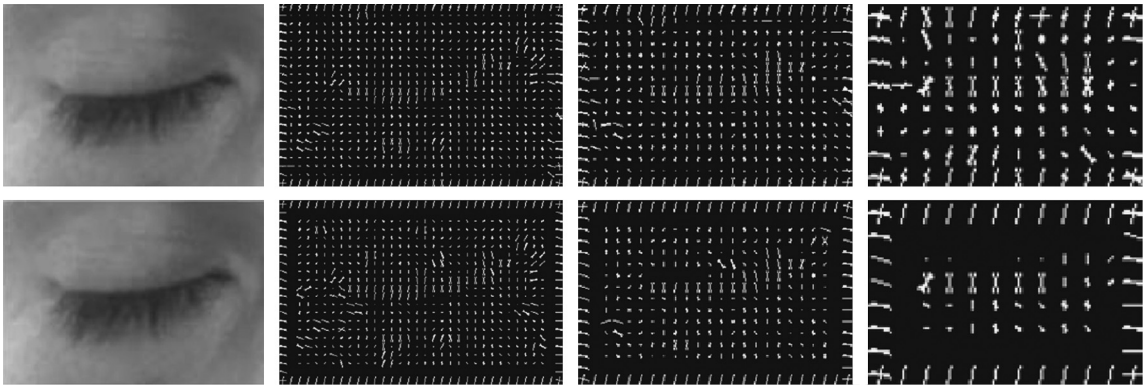

Fig. 5. Illustration of multi-scale HOG (the first row) and multi-scale HPOG (the second row). The original eye images are shown in the leftmost column, and their multi-scale representations are listed from the left to the right with different cell size: 3×3, 5×5, and 7×7 (pixels).

Let us denote the gradient values at the pixel z_i on both horizontal and vertical directions as $(g_x(z_i), g_y(z_i))$, and the corresponding magnitude and angle of the gradient as $\sqrt{g_x^2(z_i)+g_y^2(z_i)}$ and $\arctan(g_y(z_i)/g_x(z_i))$ respectively. Here, we first analyze the gradient distribution of a neighboring region and explore such information for gradient smoothing. In particular, we compute the covariance matrix C of pixels gradients on a local neighboring rectangle region $N(z_i)$ centered at the current pixel z_i as follows (assume that the gradients are normalized with zero mean):

$$C=\begin{bmatrix}\vdots & \vdots\\ g_x(z_j) & g_y(z_j)\\ \vdots & \vdots\end{bmatrix}^T\begin{bmatrix}\vdots & \vdots\\ g_x(z_j) & g_y(z_j)\\ \vdots & \vdots\end{bmatrix}$$

$$=\sum_{z_j\in N(z_i)}\begin{bmatrix}g_x^2(z_j) & g_x(z_j)g_y(z_j)\\ g_y(z_j)g_x(z_j) & g_y^2(z_j)\end{bmatrix},\quad j=1,\ldots,m \tag{1}$$

where z_j is a neighbor in $N(z_i)$ of the z_i (in our experimental setting, the local region is set as 3×3 pixels), and m is the number of pixels in $N(z_i)$. Then the largest eigenvector of the covariance matrix is used to replace the gradients of the current pixel in HOG for local shape information pooling (cf., Fig. 4). Hence the name "Histograms of Principal Oriented Gradients" (HPOG). Note that except the way to represent local shape information, the other components remain the same as the standard HOG method. Illustration of HPOG can be seen from the last column in Fig. 2.

Multi-scale HPOG (*MultiHPOG*): Another problem is related to the face detector and eye localizer – the coarsely estimated face region and eye locations may lead to eye patches with varying scales. Therefore, scale invariant features are needed for such scenarios. Geometric alignment may alleviate scales difference to some extent. Alternatively, one can estimate the scale first and then extract scale-specific features, as in Scale-Invariant Feature Transform (SIFT) [40]. However, it is unclear how to generalize this idea to HOG. Probably the easiest way to obtain the multi-scale description is to collect several patches with different scales at the same location and then extract features normally from those patches [41].

In this work, we adopt the latter method but do it at the feature level, i.e., extracting several HOG (HPOG) features by changing model parameters (using different sizes of cell or block). Local pooling size has a close relationship to the sensitivity and specificity of descriptors to the local eye region, although commonly a fixed default setting is adopted. Actually, a small cell makes the local pooling more specific but might be stuck to detailed feature changes, and vice versa. Consequently, it is difficult to choose an appropriate cell scale adaptive to each image, especially for objects with large scale changes – as mentioned before; this may be simply caused by inaccurate eye localization. Using multiple cell sizes could thus be a simple but effective way to bypass this issue. In our experiments, we use three sizes (scales) as 3×3, 5×5, and 7×7 (pixels), and set the size of each block as 2×2 (cells). The overall flow chart to extract MultiHPOG descriptor from a given image is shown in Fig. 4, and the extracted MultiHPOG descriptor from a closed eye is illustrated in Fig. 5, where we also show the multi-scale HOG descriptor for comparison.[2]

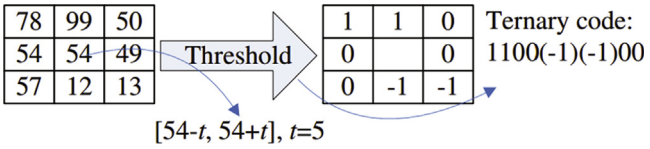

Fig. 6. Illustration of the mechanism of the LTP descriptor [30].

2.2.2. Local texture descriptor

One typical and classic local texture descriptor is the Local Binary Patterns (LBP) proposed by Ojala et al. [42], which takes a local neighborhood around each pixel, thresholds the pixels of the neighborhood at the value of the central pixel and uses the resulting binary valued image patch as a local image descriptor. LBP is resistant to lighting effects in the sense that they are invariant to monotonic graylevel transformations, and they have been shown to have high discriminative power for texture classification [42].

Local Ternary Pattern (LTP) proposed by Tan and Triggs [30] as a simple generalization of LBP has proved its improved performance. In particular, LTP extends LBP to 3-valued codes in the discretization of the difference between the central pixel and its surrounding pixels (cf. Fig. 6), where the new added value encodes the small difference in an interval of $[-t\ t]$ (t is an experiential threshold). In such a way, more detailed gradient information is explored with promising performance for tackling challenging conditions such as uneven illumination and image noise. In this study, we set the threshold t as 5, and divide each 24×24 eye patch into 3×6 blocks and represent each block as a 59-dimensional histogram. Therefore, for each eye patch we have a 3776-dimensional LTP vector ($32\times59\times2$ for the positive and negative halves of LTP coding). Illustration of LTP can be seen from the second column in Fig. 2.

[2] A Matlab implementation of both the HPOG and MultiHPOG is available at http://parnec.nuaa.edu.cn/xtan/data/ClosedEyeDatabases.html.

Please cite this article as: F. Song, et al., Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.024

2.2.3. Global shape descriptor

Gabor wavelets were originally developed to model the receptive fields of simple cells in the visual cortex and in practice, they capture a number of salient visual properties including spatial localization, orientation selectivity and spatial frequency selectivity quite well. They have been widely used in face recognition. Computationally, they are the results of convolving the image with a bank of Gabor filters with different scales and orientations, and taking the "energy image" (pixel-wise complex modulus) of each resulting output image. The most commonly used filters in face recognition have the form

$$\Psi_{\mu,\nu}(z) = \frac{\| k_{\mu,\nu} \|^2}{\sigma^2} e^{-\| k_{\mu,\nu} \|^2 \| z \|^2/(2\sigma^2)} [e^{ik_{\mu,\nu} z} - e^{-\sigma^2/2}] \quad (2)$$

where μ and ν define the orientation and scale of the Gabor kernels, respectively, $z = (x, y)$ represents the coordinate of a pixel, $\| \cdot \|$ denotes the norm operator, and the wave vector $k_{\mu,\nu}$ is defined as $k_{\mu,\nu} = k_\nu e^{i\varphi_\mu}$, where $k_\nu = k_{max}/f^\nu$ and $\varphi_\mu = \mu\pi/8$. k_{max} is the maximum frequency, and f is the spacing factor between kernels in the frequency domain [31]. We use 40 filters with eight orientations and five scales on 24×24 eye patches, then downsampling the resulting vector by 16 to a 1440-dimensional vector.

2.3. The classifiers

Classifier is an important component in the proposed architecture of the appearance-based eye closeness detection system (as shown in Fig. 3). In this work, we use the Nearest Neighbor, Support Vector Machine and Adaboost as our classifiers. The nearest neighbor method is a simple and effective nonparametric classification method and is used in this paper as our baseline.

Support Vector Machines: Support Vector Machine (SVM) is the state-of-the-art large margin classifier which has gained popularity within visual pattern recognition. One problem we should handle is the imbalance problem. That is, the number of images of closed eyes and open eyes is different,[3] which tends to increase the bias of trained SVM classifier to the class with more samples. To overcome this problem, before training, we set the penalty respective coefficients for the positive and negative samples to be $\omega_1 = (N^+ + N^-)/2N^+$, $\omega_2 = (N^+ + N^-)/2N^-$, where N^+ is the number of positive samples and N^- is the number of negative samples. We used the LIBSVM package [43] with RBF kernel for the SVM-related experiments.

Adaboost with pixel-comparison: We use the Adaboost as the second classifier to be compared. It provides a simple yet effective approach for stage-wise learning of a nonlinear classification function. In this study, we use the "difference of intensities of pixels" proposed in [44] as our features. These pixel-comparison features are efficient to compute and when combined with Adaboost, the most discriminative patterns in pixel differences between open eyes and closed eyes could be selected among a huge set of candidates (see below).

More specifically, we used five types of pixel comparison operators (and their inverses) [44]:

(1) $pixel_i > pixel_j$;
(2) $pixel_i$ within 5 units (out of 255) of $pixel_j$;
(3) $pixel_i$ within 10 units (out of 255) of $pixel_j$;
(4) $pixel_i$ within 25 units (out of 255) of $pixel_j$;
(5) $pixel_i$ within 50 units (out of 255) of $pixel_j$.

[3] In practice, it is much easier to collect images of open eyes than those of closed eyes.

The binary result of each comparison, which is represented numerically as 1 or 0, is used as features. Thus, for an eye patch of 24×24 pixels, there are $2*5*(24*24)*(24*(24-1)) = 3,179,520$ pixel-comparison features.

For feature selection, we use Adaboost while learning a strong classifier at the same time. This is done by mapping each feature to a weak classifier and then selecting the most discriminative weak classifier increasingly for an additive strong classifier. For more details, see [44]. In the training process, we investigate the influence of different number of boosted weak classifiers and also different number of candidate features for selecting the most discriminative weak classifier in each iteration, e.g., setting totally boosted classifiers as 1000 and 2000, and setting the candidate features number as 1%, 10% and 100% of all possible weak classifiers per iteration.

3. Performance evaluation

In this section, we first introduce two datasets for algorithm exploration and verification (note that other publicly available datasets, such as BioID [34], AR [35], and CAS-PEAl [36], contain face images with eyes closed as well), then go on to verify the performance of different descriptors on these datasets, under the proposed architecture described in Section 2. Finally, an investigation on the details of our method is conducted.

3.1. Data and settings

ZJU dataset: The first dataset for our experiments is collected from the ZJU Eyeblink Database [1]. There is a total of 80 video clips in the blinking video database from 20 individuals, four clips for each individual: one clip for frontal view without glasses, one clip with frontal view and wearing thin rim glasses, one clip for frontal view and black frame glasses, and the last clip with an upward view without glasses. We manually select images in each blinking process, including eye images of open, half open, closed, half closed. In addition, images of the left and the right eyes are collected separately. Some samples of the dataset are shown in Fig. 7. We can see that these images may be blurred, with low resolution or occluded by glasses.

The collected eye images are then divided into two separate sets for training and test purpose. The training set consists of images from the first 16 individuals. The test set consists of the images from the remaining 4 subjects. Note that there is no overlapping in images of subjects between the training set and the test set. To further increase the diversity of training samples, various transformations such as rotation, blurring, contrast modification, and addition of Gaussian white noise are applied to the initial set of training images, yielding about 6600 new images in total. Finally, the training set contains 7334 eye images in all, with 1574 closed eye images and 5770 open eye images. The test set is constructed with 410 closed eyes and 1230 open eyes. All these images are geometrically normalized into images of 24×24 pixels.

CEW dataset: Considering that the unconstrained real-world application scenario is full of challenging variations caused by individual difference and kinds of environment changes, including lighting, blur, occlusion, and disguise, to investigate the performance of the proposed method in these conditions, we collected a database for eye closeness detection in the wild. In particular, this dataset contains 2423 subjects, among which 1192 subjects with both eyes closed are collected directly from the Internet, and 1231 subjects with eyes open are selected from the Labeled Faces in the Wild (LFW [3]) database. Illustration of eyes images in this dataset can be seen in Fig. 8.

As mentioned before, eye patches are collected based on the coarse face region and eye position automatically and respectively

Please cite this article as: F. Song, et al., Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.024

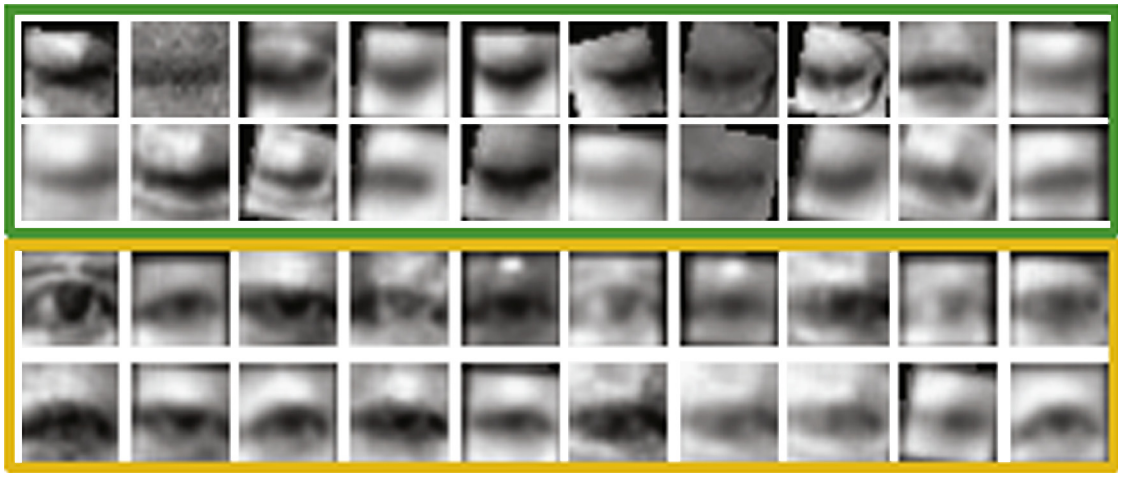

Fig. 7. Illustration of some positive (the top two rows) and negative (the bottom two rows) samples from ZJU dataset.

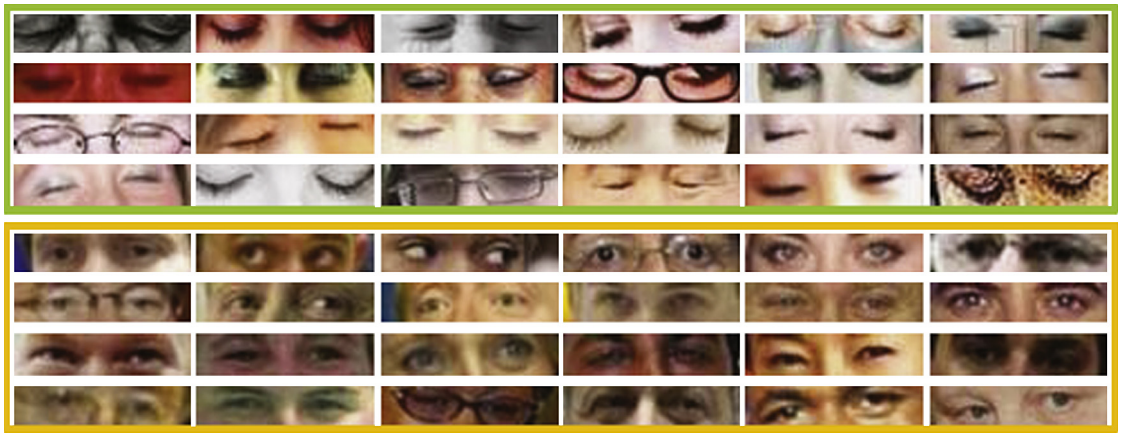

Fig. 8. Illustration of the closed eyes images (the top four rows) and the open eye images (the bottom four rows) in CEW dataset. Note that these eyes images are full of variances caused by individual, lighting, blur, occlusion, and disguise.

estimated by the face detector and eye localization. We first resize the cropped coarse faces to the size 100×100 (pixels) and then extract eye patches of 24×24 centered at the localized eye position. We randomly choose 100 faces with closed eyes and 100 faces with open eyes for training and all the remaining for testing (i.e., 2223 faces), and repeat such process ten times for mean performance reporting. Since there is great similarity between the left and right eyes, this strategy ensures that the left and right eyes from the same subject cannot be appeared in the training set and the test set simultaneously (e.g., one for training and the other for testing).

The final distributed datasets consist of the grayscale eye patches and the corresponding color face images.[4] For converting a color image into its grayscale counterpart, we eliminate its hue and saturation information while retaining its luminance (equal to the weighted sum of the R, G, and B components: 0.2989*R + 0.5870*G + 0.1140*B).

3.2. Extensive experimental results

3.2.1. Performance comparison among various feature sets

In this section, we first make a comparison between feature-based methods and appearance-based methods for eye closeness detection. One representative method of the former is the intensity projection method. Such method usually exploits heuristics that open eyes usually have a low brightness area in the eye center adjacent to two high brightness areas, corresponding to the iris and the eye whites respectively. Given a M by N image $I_{M\times N}$, a vertical projection curve (a vector) is calculated by accumulating gray values of pixels in each column and defined as $\sum_{i=1}^{M} I(i,j)$, where i is the row number, and j is the column number. Recent work [8] proposed a set of rules for judging eye states according to the shape of the vertical projection curve. In particular, for an open eye, the location of the valley (indicated as "Xmin" in Fig. 9) should at the center position around, and the value of the minimal adjacent summit (indicated as "Ysmax" in Fig. 9) should be higher enough than that of the valley (indicated as "Ymin" in Fig. 9); otherwise it is decided to be a closed eye. We implement the method in [8] and test it on our two datasets.

Fig. 10 gives the results. We can see that this method achieves performances of 77.8% and 70.1% on ZJU dataset and CEW dataset respectively. It can also be observed that its performance lag largely behind the appearance-based methods. To gain more understanding on this, we give some typical errors made by it in Fig. 9. It can be seen that poor imaging conditions, such as low resolution, blur, uneven light, may lead to ambiguous appearance of eyes, and this in turn results in the shape deterioration of the projection curve. This also explains why its performance on CEW dataset is poorer than that on ZJU dataset.

Next, we compare the performance among various descriptors mentioned in Section 2.2 with the same settings (with SVM

[4] Publicly available at http://parnec.nuaa.edu.cn/xtan/data/ClosedEyeDatabases.html.

Please cite this article as: F. Song, et al., Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.024

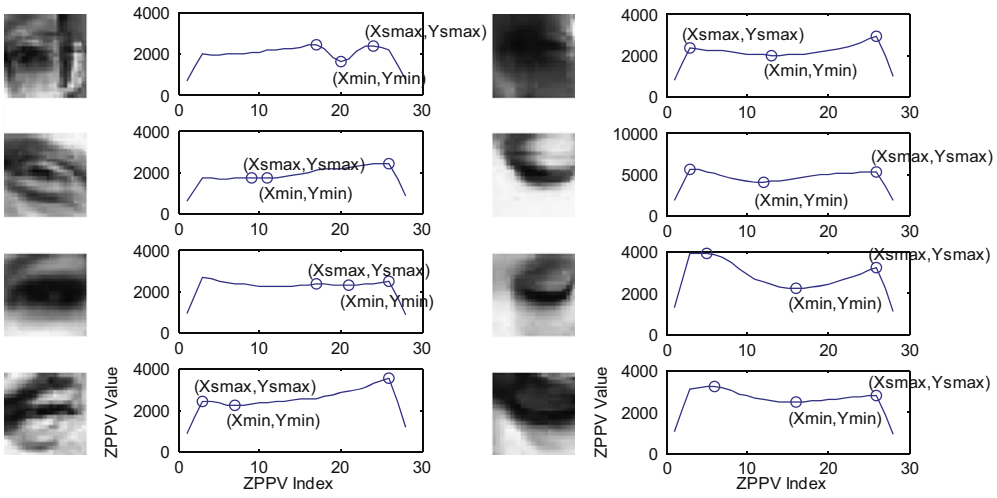

Fig. 9. Deformed vertical projection curves lead to incorrect eye state estimation by [8] due to challenging imaging conditions. The left two columns show four typical false-positive samples (they are actually open eyes, but are judged as closed eyes by the system) and their projection curves, and the right two columns show four typical false-negative eyes and their projection curves, respectively.

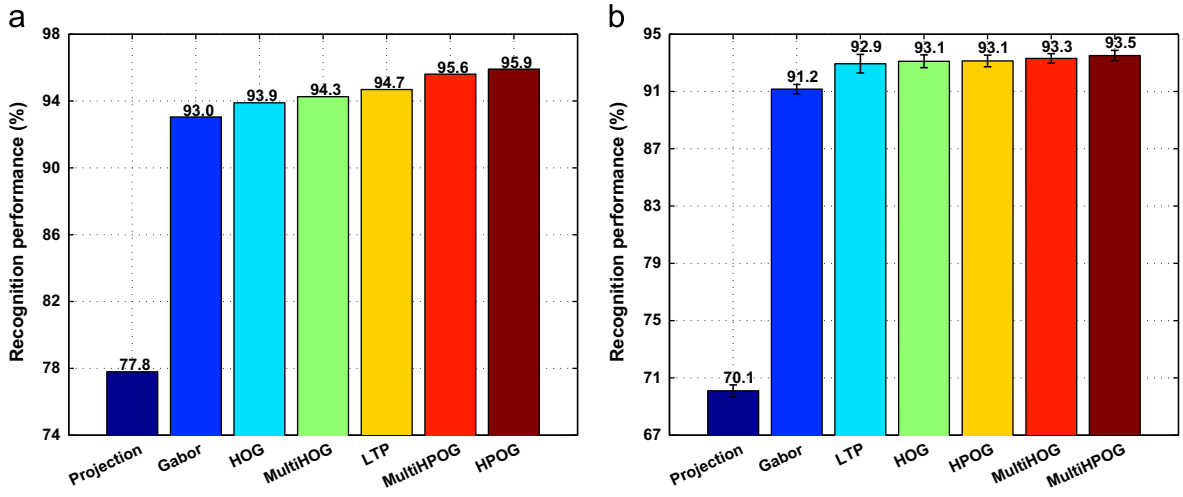

Fig. 10. Comparative accuracy of different kinds of features (a) on ZJU dataset; (b) on CEW dataset.

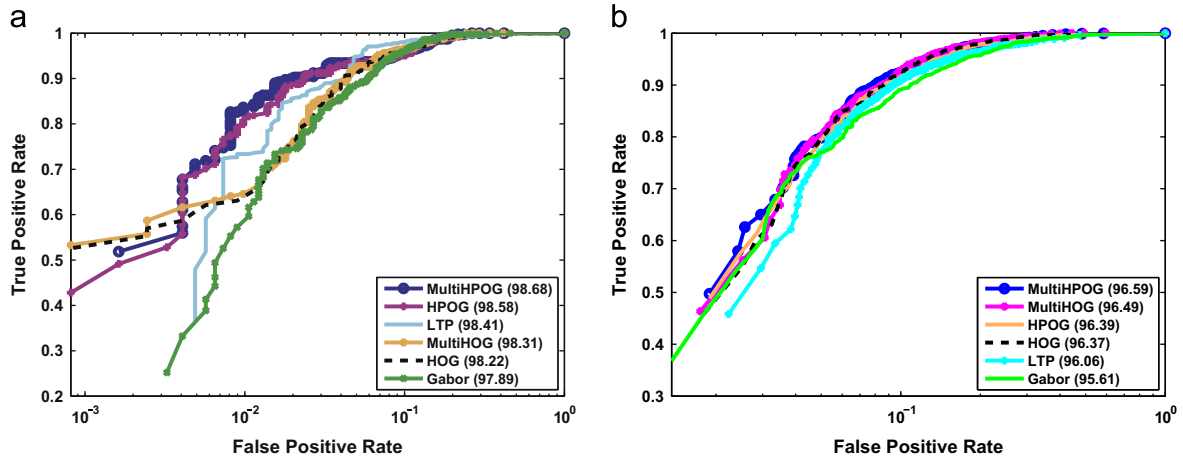

Fig. 11. ROC curves of various features using the SVM classifier (a) on ZJU dataset; (b) on CEW dataset. AUC values are given at the end of corresponding legend texts.

classifier) and the same datasets. Fig. 10 gives results. We can see that the best performers on ZJU dataset and CEW dataset are the newly proposed HPOG and MultiHPOG feature sets, respectively. This indicates that the proposed two extended HOG features are effective in extracting robust local shape features for the task of eye closeness detection. LTP feature sets perform better than both the original HOG

Please cite this article as: F. Song, et al., Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.024

feature and the Gabor features on ZJU dataset. Fig. 11 gives the ROC (Receiver Operating Characteristic) curves of those features on the two datasets. It can be seen that the MultiHPOG feature performs best in terms of AUC (Area Under the ROC Curve) values on both datasets, followed by HPOG, which further verifies that our extensions to HOG are beneficial to this task.

Table 1 gives a comprehensive performance comparison of various feature sets in terms of recognition accuracy, AUC, Equal Error Rate (EER), and also the time cost (millisecond for testing each eye patch). Several observations can be made from this table.

1. We can see that LTP obtains the lowest EER value of 5.04 on ZJU dataset, while the Gabor feature performs worst (except the gray feature) with the highest EER values of 7.16 and 10.31 on the two datasets, respectively. One possible explanation is that the LTP features give a detailed account of the appearance of eye regions while being insensitive to the lighting changes. Actually, when one screws up his eyes, it is difficult to make a decision on whether his eyes are closed with a coarse account of the global shape information (as Gabor features do). In such cases, it would be better to look at feature sets capturing the local texture or the local shape information.
2. One may notice from Fig. 11(b) that, with deteriorating image quality (e.g., those in CEW dataset), the difference between various feature sets becomes less evident. This may imply that there is a performance boundary of each feature set in dealing

Table 1
Comparative performance of various features and classifiers on the two datasets.

Approach	ZJU dataset				CEW dataset			
	Acc. (%)	AUC (%)	EER (%)	Time (ms)	Accu. (%)	AUC (%)	EER (%)	Time (ms)
NN								
Gray	84.74	–	–	0.50	74.31 ± 0.85	–	–	0.04 ± 0.01
LBP	89.19	–	–	1.59	81.00 ± 0.97	–	–	0.85 ± 0.01
LTP	91.39	–	–	3.36	83.59 ± 1.45	–	–	1.64 ± 0.02
Gabor	85.04	–	–	14.87	85.53 ± 1.22	–	–	13.79 ± 0.02
HOG	90.90	–	–	7.76	90.35 ± 0.38	–	–	6.82 ± 0.01
MultiHOG	90.72	–	–	13.79	90.47 ± 0.39	–	–	12.67 ± 0.02
HPOG	90.42	–	–	13.57	89.93 ± 0.42	–	–	12.70 ± 0.01
MultiHPOG	90.72	–	–	32.59	90.50 ± 0.46	–	–	30.38 ± 0.02
SVM								
Gray	89.62	95.92	11.55	1.68	82.85 ± 1.08	89.48 ± 0.81	18.38 ± 1.06	0.32 ± 0.02
LBP	94.51	98.17	5.37	4.05	91.12 ± 0.50	95.14 ± 0.40	10.61 ± 0.57	1.96 ± 0.06
LTP	94.69	98.41	**5.04**	12.14	92.94 ± 0.65	96.06 ± 0.40	8.83 ± 0.64	16.67 ± 1.22
Gabor	93.04	97.89	7.16	16.85	91.16 ± 0.33	95.61 ± 0.24	10.31 ± 0.38	17.65 ± 0.84
HOG	93.89	98.22	6.27	12.24	93.10 ± 0.41	96.37 ± 0.24	8.69 ± 0.31	12.61 ± 0.43
MultiHOG	94.26	98.31	6.35	19.08	93.31 ± 0.34	96.49 ± 0.22	8.52 ± 0.39	19.81 ± 0.51
HPOG	**95.91**	98.58	6.18	18.17	93.13 ± 0.45	96.39 ± 0.21	8.60 ± 0.42	18.55 ± 0.29
MultiHPOG	95.60	**98.68**	6.27	37.57	**93.51** ± 0.36	**96.59** ± 0.21	**8.17** ± 0.39	38.47 ± 0.53
Ada. (1000)								
1%	92.06	96.48	8.30	**0.034**	87.09 ± 0.86	94.28 ± 0.52	12.98 ± 0.85	0.029 ± 0.003
10%	92.91	96.52	8.71	0.036	87.01 ± 0.76	94.14 ± 0.57	13.02 ± 0.82	0.028 ± 0.002
100%	92.31	96.50	8.45	0.042	86.74 ± 0.86	94.08 ± 0.61	13.32 ± 0.83	**0.025** ± 0.002
Ada. (2000)								
1%	92.37	96.67	7.65	0.063	87.74 ± 0.76	94.50 ± 0.54	12.21 ± 0.89	0.060 ± 0.009
10%	92.67	96.73	7.48	0.063	86.70 ± 0.71	94.07 ± 0.59	13.07 ± 0.69	0.048 ± 0.002
100%	93.47	97.01	7.08	0.065	86.66 ± 0.65	93.97 ± 0.66	13.47 ± 0.71	0.045 ± 0.002

Table 2
Comparative performance of various feature fusion schemes (with the SVM classifier) on the two datasets. "(s)" indicates feature fusion at the score level otherwise at the feature level.

Approach	ZJU dataset			CEW dataset		
	Acc. (%)	AUC (%)	EER (%)	Acc. (%)	AUC (%)	EER (%)
LTP+Gabor	95.54	98.81	4.88	94.01 ± 0.65	96.75 ± 0.30	7.71 ± 0.64
LTP+Gabor(s)	95.48	98.77	4.88	93.89 ± 0.61	95.00 ± 0.69	7.94 ± 0.53
HPOG+LTP	96.40	99.01	4.56	93.94 ± 0.43	96.71 ± 0.23	7.89 ± 0.39
HPOG+LTP(s)	95.91	98.66	4.80	94.13 ± 0.39	95.49 ± 0.40	7.79 ± 0.44
HPOG+Gabor	96.58	99.03	4.56	93.91 ± 0.39	96.72 ± 0.17	7.93 ± 0.35
HPOG+Gabor(s)	95.18	98.52	5.78	93.66 ± 0.39	95.61 ± 0.50	7.90 ± 0.41
MultiHPOG+LTP	96.46	99.14	4.23	93.94 ± 0.35	96.72 ± 0.21	7.90 ± 0.41
MultiHPOG+LTP(s)	96.21	98.65	4.23	94.31 ± 0.44	95.61 ± 0.43	7.62 ± 0.45
MultiHPOG+Gabor	96.40	98.88	5.37	93.82 ± 0.32	96.72 ± 0.20	8.06 ± 0.30
MultiHPOG+Gabor(s)	95.79	98.66	5.13	94.03 ± 0.46	95.72 ± 0.61	7.70 ± 0.37
HPOG+LTP+Gabor	96.64	99.04	3.99	94.41 ± 0.44	96.93 ± 0.19	7.48 ± 0.46
HPOG+LTP+Gabor(s)	95.91	89.22	3.58	94.54 ± 0.54	96.15 ± 0.40	7.33 ± 0.50
MultiHPOG+LTP+Gabor	**96.83**	**99.27**	**3.09**	94.45 ± 0.48	**96.94** ± 0.19	7.47 ± 0.40
MultiHPOG+LTP+Gabor(s)	96.40	96.67	4.64	**94.72** ± 0.48	95.19 ± 0.40	**7.26** ± 0.47

Please cite this article as: F. Song, et al., Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.024

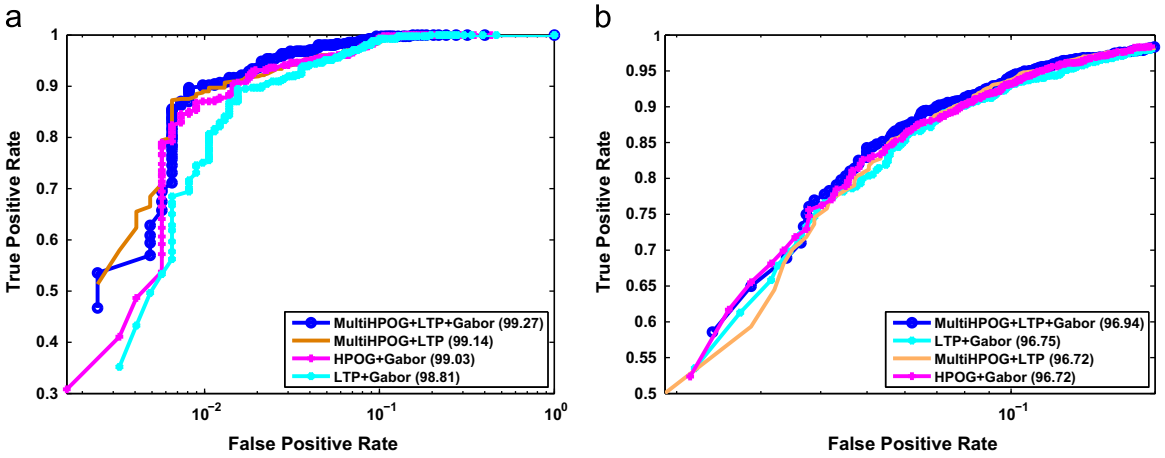

Fig. 12. ROC curves of various feature fusion strategies using the SVM classifier (a) on ZJU dataset; (b) on CEW dataset. AUC values are given at the end of corresponding legend text.

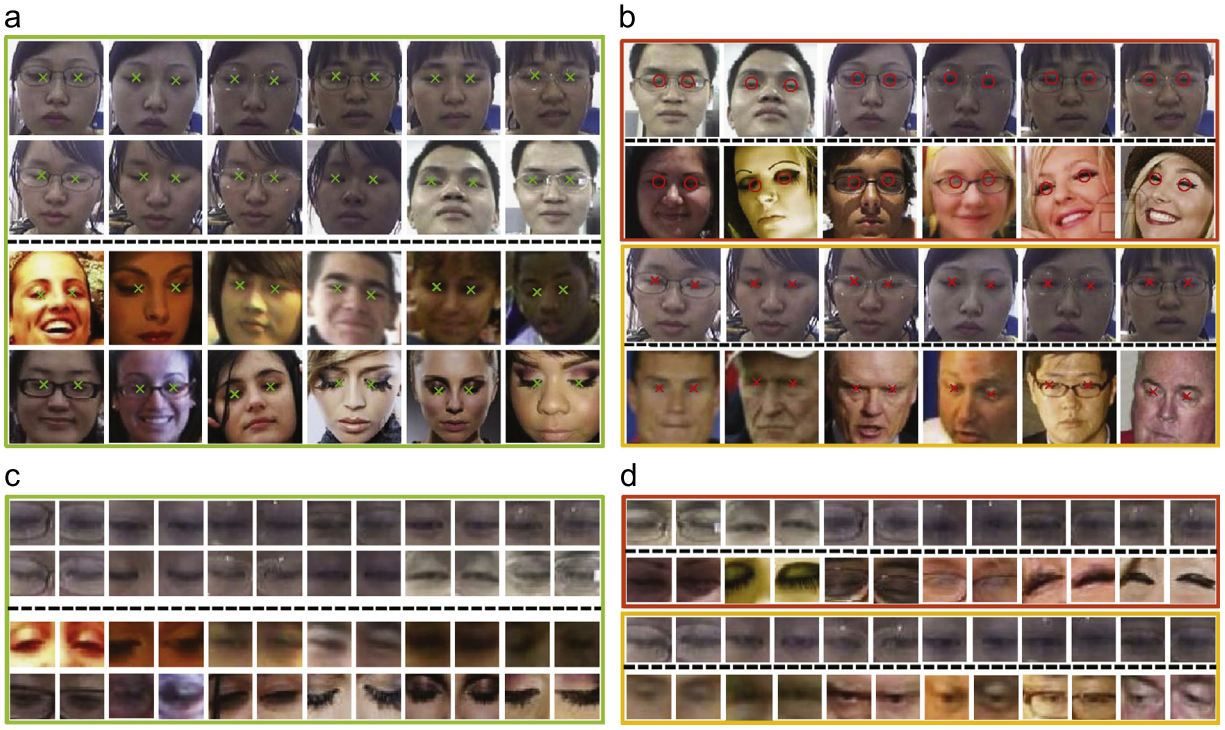

Fig. 13. Examples of some typical success and failure cases of our method. For each face image shown in the top four rows ((a) and (b)), its eye locations are automatically located and are marked as cross signs, based on which, the eye patches are extracted and are illustrated in the bottom four rows ((c) and (d)). All the eye states of the images on the left side ((a) and (c)) are successfully recognized while those on the right side ((b) and (d)) are failed. The images in the upper two rows of (b) and (d) illustrate the false-negative cases (i.e., the system regards the closed eyes as the open ones, marked by the red circles), and those in the lower two rows give the false-positive examples (i.e., the open eyes are incorrectly recognized as the closed eyes, marked by the red crosses). For all the images in this figure, those above the dashed line are from ZJU dataset and below it from CEW dataset. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

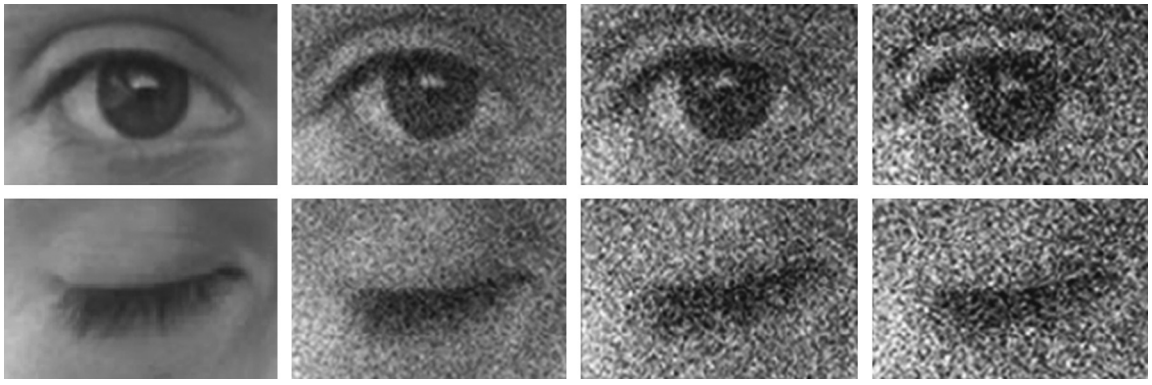

Fig. 14. Illustration of eye images contaminated by different degrees of Gaussian noise (from left to right the variance are 0, 0.01, 0.03, and 0.05).

with images with low quality. One way to address this issue is to fuse the (weak) information captured by different feature sets, as described in the next section.

3. Although the SVM-based methods outperform the Adaboost-based ones in terms of recognition accuracy, they run much slower than the latter ones at the test time. In particular, one can see from the table that the Adaboost-based methods run at least 100 times faster than SVM without hurting the ROC performance (AUC values) too much. This suggests that the Adaboost with the difference of pixel features is a very attractive candidate in practice, especially in cases where real time test is of importance.

3.2.2. *Fusion feature sets for difficult images*

To investigate the possible benefits of feature fusion for images under uncontrolled conditions, we conducted a series of experiments on both ZJU dataset and CEW dataset by fusing various descriptors. Feature fusion could be performed either at the feature level or the score level, and we tested both in this work. Table 2 gives the overall results. Comparing this with the results shown in Table 1, one can see that fusing features improves performance in general. For example, on ZJU dataset, the accuracy of Gabor wavelets and LTP is 93.04% and 94.69%, respectively, while combining them at the feature level improves this to 95.54%. For HPOG features, its performance improves from 95.91% to 96.40% when combined with the LTP features, which is further improved to 96.64% if the Gabor features are added. Similar results can be observed on CEW dataset, as well. For example, combining Gabor and LTP features improves the performance to 94.01% from 91.16% and 92.94%, respectively. Overall, fusing the MultiHPOG, LTP, and Gabor features gives the best result on both ZJU dataset and CEW dataset. Further details of ROC curves of feature fusion strategies are given in Fig. 12.

Fig. 13(a) and (c) illustrates several typical successful examples of our "MultiHPOG / LTP / Gabor + SVM" method. One can see that although those images are taken under uncontrolled conditions and their appearance changes largely, our method correctly identifies these eyes as closed. In addition, Fig. 13(b) and (d) gives some representative failure cases, including both the false-positive and false-negative ones. From those images, one can see that they look even confusing to human beings when deciding whether these eyes are closed. This helps us to understand the challenges of

Table 3
Comparative accuracy (%) of HPOG and HOG at different Gaussian noise levels on the two datasets.

Dataset and features	Noise level		
	0.01	0.03	0.05
ZJU dataset			
HOG	83.7 ± 0.9	77.3 ± 0.6	76.0 ± 0.5
HPOG	**86.2** ± 0.6	**80.2** ± 0.5	**78.1** ± 0.4
CEW dataset			
HOG	80.4 ± 0.3	68.1 ± 0.7	63.1 ± 0.8
HPOG	**82.9** ± 0.6	**70.8** ± 0.8	**65.6** ± 0.7

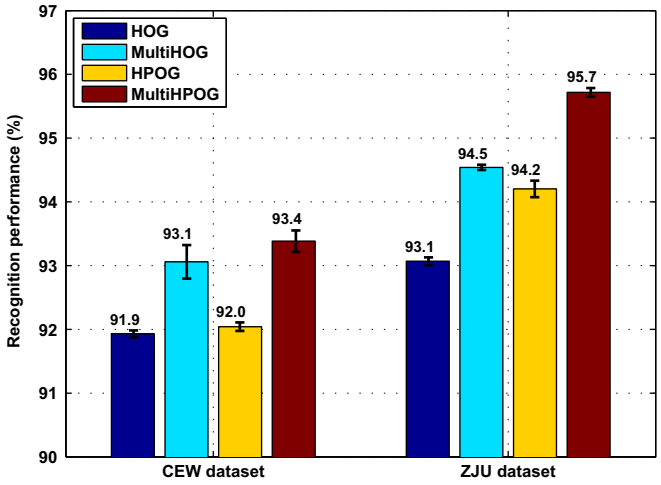

Fig. 16. Comparative accuracy of Multi-scale HOG and HPOG on ZJU dataset and CEW dataset.

Fig. 15. Illustration of eye patches with varying scales. The sizes of eye patches are 24, 30, and 36 from left to right.

Please cite this article as: F. Song, et al., Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.024

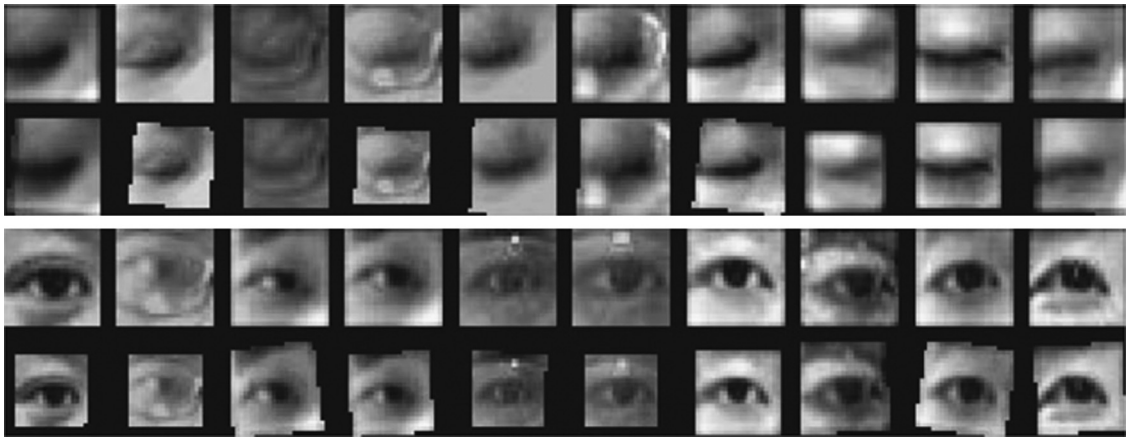

Fig. 17. Illustration of eye patches normalized with the congealing method [38], where patches in the top two rows are original images of closed eyes and their corresponding normalized versions respectively, and patches in the bottom two rows are original images of open eyes and their corresponding normalized images, respectively.

eye closeness detection in the real world and further implies the need of further research on these.

3.2.3. The robustness of HPOG against Gaussian noise

To verify the capability of the proposed HPOG descriptor against image noise, we simulate the noise imaging conditions by adding varying degrees of Gaussian noise to eye images (Fig. 14), and then extract the HPOG feature sets from them. For the two datasets, we follow the same experimental protocol for training and testing data splitting as introduced previously with the SVM as the classifier, and repeat the Gaussian noise process ten times and report the mean performance with standard variance. Table 3 shows the results. It can be observed that the performance of both HOG and HPOG decreases with the increasing Gaussian noise level, which shows that Gaussian noise affects the stableness of gradient information extraction. However, one can also see that the HPOG outperforms HOG by about 2% at each Gaussian noise level consistently on the two datasets. This indicates that it is beneficial to use more robust local shape estimators when images contain noise.

3.2.4. Multi-scale extensions of HOG and HPOG

To investigate the effectiveness of the proposed multi-scale extensions of both HOG and HPOG, we first simulate the multi-scale scenario by collecting eye patches in three scales, i.e., 24×24, 30×30, and 36×36 (see Fig. 15), and then resize them to the same size of 24×24 pixels. For CEW dataset, we randomly select 200 subjects with one image per person (100 with closed eyes and 100 with open eyes) for classifier training. This results in $200(\textit{subjects})\times 3(\textit{scales})\times 2(\textit{eyes per subject})=1200$ eye patches, and eye patches from the remaining 2223 faces are used for testing. Similarly, for ZJU dataset, 200 faces are randomly chosen from the first 16 individuals, and the faces of the remaining 4 individuals are used for testing. Such processes are repeated 10 times for mean performance reporting with standard variance. Fig. 16 gives the results. We can see from the figure that both multi-scale HOG and multi-scale HPOG improve the performance upon their original version. For example, on ZJU dataset, the improvement of multi-scale version of HOG and HPOG is respectively 1.4% and 1.5%, compared to the original one, while, on CEW dataset, the improvement is 1.2% and 1.4% respectively.

3.2.5. The importance of eye patch alignment

Finally, we investigate the influence of the eye patch alignment on the performance of the system. Fig. 17 illustrates the aligned eye patches. Although the LTP feature is known to be rotation-invariant, and the HOG feature is robust against a slight perturbation in the image, they are not robust against general affine transformations. Indeed, Table 4 shows that it is beneficial to do geometric normalization for eye patches before extracting features from them.

Table 4
Comparative accuracy of typical features with/without (w/o) eye alignment (with the SVM classifier) on ZJU dataset.

Processing	LTP	Gabor	HOG	HPOG
Without alignment	94.3	89.23	89.5	93.04
With alignment	**94.7**	**93.0**	**93.9**	**95.9**

3.3. Runtime performance evaluation

We evaluate the runtime performance of our approach on video sequences of the ZJU Eyeblink database [1]. In particular, on a laptop, we run the whole pipeline of our eye state detection (cf. Fig. 3) on each frame of a randomly selected video sequence consisting of 148 frames.[5] Table 5 reports the average detection speed and the detailed elapsed time per frame by each processing step. The table reveals that the step of feature extraction takes about 28.5% of the total time, while 62.9% of the time is due to the preprocessing step. In practice, we may replace the preprocessing step with a suitable eye tracking algorithm to save the time spent on the steps of face detection and/or eye localization, and use a more efficient patch alignment algorithm. Note that our implementation is based on the Matlab platform without any code optimization and the actual runtime efficiency could be further improved by using other low-level programming language such as C and by using code optimization techniques.

3.4. Comparison with other methods

Due to the lack of common datasets and widely accepted evaluation protocol, it is really hard to make a fair comparison between our method and other methods. Despite this, Table 6 lists some of the methods we are aware of, with corresponding experimental settings such as the dataset tested on, major characteristics of

[5] We also test several other sequences of this database, and they give similar results.

Please cite this article as: F. Song, et al., Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.024

Table 5
Typical time cost (ms/frame) for each step of the proposed method for eye closeness detection on the ZJU Eyeblink database [1].

Preprocessing (ms)			Feature extraction (ms)			Prediction (ms)	Total (ms)
Face detection	Eye localization	Alignment	LTP	Gabor	MultiHPOG		
13	11	180	5	21	67	29	326

Table 6
Comparison of some state-of-the-art appearance-based methods for eye closeness detection.

Approach	**Dataset**	**Challenge**	**#Train:** open (o) closed (c)	**#Test:** open (o) closed (c)	**Eye size**	**Acc. (%)**
LBIIPH + Adaboost [23]	Mixed data from CAS-PEAL [36], RPI ISL eyes [45], AR [35] and BioID [34]	Varying sizes, skin colors, orientations, illuminations	2000(o) 1000(c)	2979(o) 1479 (c)	40×20	99.84
LBP + SVM [21]	CAS-PEAL [36]	Variations in expression, background, accessory	200(o) 200(c)	5738(o) 552(c)	40×20	96.50
Color correlogram + Adaboost [29]	WebCam data	Varying illumination	1500(o) 700(c)	1618(o) 812(c)	60×30	98.39
Gabor + SVM [25]	BioID [34]	Varying skin color, illumination, gender	738(o) 316(c)	300(o) 100(c)	40×20	94.00
Gabor + SVM [26]	Surveillance data	Varying illuminations, races, eye colors	–	2810(o) 1280 (c)	30×20	95.59
LBP + SVM [22]	ZJU eyeblink dataset [1]	varying pose, lighting, accessory	Front clips Clips type 1, 2	Rear clips Clips type 3, 4	0.74×0.37 (ratio to eyes distance)	90.37 84.32
Haar + Adaboost [28]	Web data	Varying resolution	9000(o) 7400(o)	5754	24×24	94.71
MultiHPOG + LTP + Gabor + SVM (ours)	ZJU dataset	Variations in pose, lighting, accessory	5770(o) 1570(c)	1230(o) 410(c)	24×24	96.83
	Web data	Variations in lighting, blur, occlusion, disguise	200(o) 200(c)	2262(o) 2184 (c)		94.72
	BioID [34]	Varying skin color, illumination, gender	5770(o) 1570(c) (ZJU dataset)	2910(o) 132(c)		97.14
	CAS-PEAL [36]	Variations in expression, background, accessory		7292(o) 782(c)		99.05
	AR [35]	Variations in lighting, expression		3028(o) 512(c)		99.75

the dataset, training/test data partitions, and the performance. From the table, we can see that while many of these methods were evaluated on various datasets such as BioID [34], AR [35], CAS-PEAL [36], what they all have in common is that the feature descriptors adopted for eye representation are similar to or as same as those mentioned in Section 2.2, such as LBP, Gabor, Haar, or their variants. It can be seen that our method shows an advantage to the LBP-based method [22] which obtains a performance of 90.37% on the ZJU Eyeblink dataset [1]. This performance advantage may be attributed to the proposed robust MultiHOPG descriptor and the multi-feature fusion strategy. Both [28] and our methods are tested under totally uncontrolled conditions with a similar performance. However, our "MultiHPOG+LTP+Gabor" feature fusion method is trained on a much smaller dataset than that in [28] (about a 40-time reduction in terms of the size of the training set).

To further verify the effectiveness of the proposed method, besides ZJU dataset, we also conduct a series of experiments on several public face datasets, including BioID [34], CAS-PEAL [36], and AR [35]. Particularly, we run our closeness detector trained on ZJU dataset directly on these datasets, without any fine-tuning on them, and report the accuracy according to the manually labeled ground truth.[6] The last three rows of Table 6 give the detailed experimental settings and the corresponding results. Note that since the definition of ground truth is different on these tested datasets by different authors, results listed in Table 6 are not directly comparable. Nevertheless, it is clear that our method gives excellent detection accuracy on the AR and the CAS-PEAL dataset, and slightly worse results on the BioID. This demonstrates the generalization capability of the proposed approach since we did not train on these datasets but only test our model on them.

It is worth mentioning that Pan et al. [1] has reported eye-blink detection rate on the ZJU Eyeblink dataset.[7] Particularly, they achieved a two-eye-blink detection rate of 93.3% (84.1% for one eye), further improved to 95.7% (88.8% for one eye) by taking contextual eye state information into account. This implies that eye blink detection could be made tolerant to the mis-recognition of individual eye state (such as eye closeness), benefiting from the dynamic information provided by the nearby state. On the other hand, we obtain accuracy of 96.8% for eye closeness detection on ZJU dataset without using any dynamic contextual information. It will be interesting to investigate whether the performance of an eye-blink detector (an activity recognizer, actually) could be further improved if an enhanced individual eye state detector like ours is used as its component. However, this is beyond the scope of the current paper and will be the focus of our future work.

[6] The labels are available at http://parnec.nuaa.edu.cn/xtan/data/ClosedEyeDatabases.html.

[7] Note that this dataset is different from the ZJU dataset used in this work – the latter is specially designed for eye closeness detection on the basis of the former.

Please cite this article as: F. Song, et al., Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.024

4. Conclusion

Eye closeness detection has wide applications in practice including fatigue detection and anti-blink system with cameras, but this problem is far from being solved, especially when the images are captured from uncontrolled real-world scenarios. In this work, we first investigate several typical feature descriptors to understand their respective capability of distinguishing closed eyes from open ones, and find that the HOG-type descriptors (i.e., the new features that we have proposed, HPOG and its Multi-scale version named MultiHPOG) and LTP descriptors are competent even when the quality of eye images is decreasing (as shown in Fig. 8), and MultiHPOG is the best individual feature in our experiments.

We also evaluate different combinations of features in our experiments and find that local texture (LTP) and local shape (MultiHPOG) features are important, and adding the global shape feature (Gabor wavelets) further improves the performance. Furthermore, we show that the geometric normalization and feature extraction at multiple scales are of importance in dealing with the issues caused by the inaccurate eye localization.

Last but not least, considering that one major reason that most current work on eye closeness detection could not be directly comparable is due to the lack of common datasets for evaluation, we contribute two new large eye-state databases in this regard. We are currently working on more effective descriptors and more efficient detection model under the proposed appearance-based eye closeness recognition architecture.

Conflict of interest statement

None declared.

Acknowledgments

The authors are grateful to the editors and reviewers for helpful comments and suggestions. This work was supported by the National Science Foundation of China (61073112, 61035003, 61373060), Jiangsu Science Foundation (BK2012793), Qing Lan Project, Research Fund for the Doctoral Program (RFDP) (20123218110033), and the Fundamental Research Funds for the Central Universities (CXLX11_0204).

Appendix A. Supplementary material

Supplementary data associated with this paper can be found in the online version at http://dx.doi.org/10.1016/j.patcog.2014.03.024.

References

[1] G. Pan, L. Sun, Z. Wu, S. Lao, Eyeblink-based anti-spoofing in face recognition from a generic webcamera, in: Proceedings of IEEE International Conference on Computer Vision, 2007, pp. 1–8.
[2] F. Song, X. Tan, S. Chen, Z.-H. Zhou, A literature survey on robust and efficient eye localization in real-life scenarios, Pattern Recognit. 46 (12) (2013) 3157–3173.
[3] G.B. Huang, M. Ramesh, T. Berg, E. Learned-Miller, Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments, Technical Report 07-49, University of Massachusetts, Amherst, October 2007.
[4] J. Orozco, F. Roca, J. Gonzàlez, Real-time gaze tracking with appearance-based models, Mach. Vis. Appl. 20 (2009) 353–364.
[5] Y. Feng, D. Hu, P. Ning, A combined eye states identification method for detection of driver fatigue, in: Proceedings of IET International Communication Conference on Wireless Mobile and Computing, 2009, pp. 217–220.
[6] P.R. Tabrizi, R.A. Zoroofi, Drowsiness detection based on brightness and numeral features of eye image, in: Proceedings of International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2009, pp. 1310–1313.
[7] A. Liu, Z. Li, L. Wang, Y. Zhao, A practical driver fatigue detection algorithm based on eye state, in: 2010 Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics (PrimeAsia), 2010, pp. 235–238.
[8] M. Dehnavi, M. Eshghi, Design and implementation of a real time and train less eye state recognition system, EURASIP J. Adv. Signal Process. 2012 (1) (2012) 1–12.
[9] A. Królak, P. Strumiłło, Eye-blink detection system for human–computer interaction, Univers. Access Inf. Soc. 11 (4) (2012) 409–419.
[10] W.O. Lee, E.C. Lee, K.R. Park, Blink detection robust to various facial poses, J. Neurosci. Methods 193 (2) (2010) 356–372.
[11] K. Arai, R. Mardiyanto, Real time blinking detection based on Gabor filter, Int. J. Hum. Comput. Interact. 1 (3) (2010) 33–40.
[12] Q. Wang, J. Yang, Eye location and eye state detection in facial images with unconstrained background, J. Inf. Comput. Sci. 1 (5) (2006) 284–289.
[13] H. Tan, Y. Zhang, Detecting eye blink states by tracking iris and eyelids, Pattern Recognit. Lett. 27 (6) (2006) 667–675.
[14] T. D'Orazio, M. Leo, C. Guaragnella, A. Distante, A visual approach for driver inattention detection, Pattern Recognit. 40 (8) (2007) 2341–2355.
[15] H. Noor, R. Ibrahim, A framework for measurement of human's fatigue level using 2 factors, in: Proceedings of International Conference on Computer and Communication Engineering, 2008, pp. 414–418.
[16] Y. Lei, M. Yuan, X. Song, X. Liu, J. Ouyang, Recognition of eye states in real time video, in: Proceedings of International Conference on Computer Engineering and Technology, vol. 1, 2009, pp. 554–559.
[17] M. Eriksson, N. Papanikotopoulos, Eye-tracking for detection of driver fatigue, in: Proceedings of IEEE Conference on Intelligent Transportation System, 1997, pp. 314–319.
[18] Z. Zhang, J. Zhang, Driver fatigue detection based intelligent vehicle control, in: Proceedings of IEEE International Conference on Pattern Recognition, vol. 2, 2006, pp. 1262–1265.
[19] M.S. Devi, P.R. Bajaj, Driver fatigue detection based on eye tracking, in: Proceedings of International Conference on Emerging Trends in Engineering and Technology, 2008, pp. 649–652.
[20] L. Lu, X. Ning, M. Qian, Y. Zhao, Close eye detected based on synthesized gray projection, in: Advances in Multimedia, Software Engineering and Computing, vol. 2, 2012, pp. 345–351.
[21] R. Sun, Z. Ma, Robust and efficient eye location and its state detection, Adv. Comput. Intell. 23 (2009) 318–326.
[22] Y. Wu, T. Lee, Q. Wu, H. Liu, An eye state recognition method for drowsiness detection, in: Proceedings of Vehicular Technology Conference, 2010, pp. 1–5.
[23] L. Zhou, H. Wang, Open/closed eye recognition by local binary increasing intensity patterns, in: Proceedings of IEEE Conference on Robotics, Automation and Mechatronics, 2011, pp. 7–11.
[24] Y.-L. Tian, T. Kanade, J. Cohn, Eye-state action unit detection by Gabor wavelets, in: Proceedings of International Conference on Advances in Multimodal Interfaces, 2000, pp. 143–150.
[25] E. Cheng, B. Kong, R. Hu, F. Zheng, Eye state detection in facial image based on linear prediction error of wavelet coefficients, in: Proceedings of IEEE International Conference on Robotics and Biomimetics, 2009, pp. 1388–1392.
[26] M.J. Flores, J.M. Armingol, A. de la Escalera, Driver drowsiness warning system using visual information for both diurnal and nocturnal illumination conditions, In: EURASIP Journal of Advance Signal Process Special Title on Vehicular ad hoc Networks 2010, 2010, Article No. 3. 〈http://dx.doi.org/10.1155/2010/438205〉.
[27] H.Y. Yang, X.H. Jiang, L. Wang, Y.H. Zhang, Eye statement recognition for driver fatigue detection based on Gabor wavelet and HMM, Appl. Mech. Mater. 128 (2012) 123–129.
[28] Z. Liu, H. Ai, Automatic eye state recognition and closed-eye photo correction, in: Proceedings of International Conference on Pattern Recognition, 2008, pp. 1–4.
[29] H. Wang, L. Zhou, Y. Ying, A novel approach for real time eye state detection in fatigue awareness system, in: Proceedings of Robotics Automation and Mechatronics, 2010, pp. 528–532.
[30] X. Tan, B. Triggs, Enhanced local texture feature sets for face recognition under difficult lighting conditions, IEEE Trans. Image Process. 19 (6) (2010) 1635–1650.
[31] M. Lades, J. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R. Wurtz, W. Konen, Distortion invariant object recognition in the dynamic link architecture, IEEE Trans. Comput. 42 (3) (1993) 300–311.
[32] P. Viola, M. Jones, Robust real-time face detection, Int. J. Comput. Vis. 57 (2007) 137–154.
[33] X. Tan, F. Song, Z. Zhou, S. Chen, Enhanced pictorial structures for precise eye localization under uncontrolled conditions, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1621–1628.
[34] O. Jesorsky, K.J. Kirchberg, R.W. Frischholz, Robust face detection using the Hausdorff distance, in: Proceedings of Audio-and Video-based Biometric Person Authentication, 2001, pp. 90–95.
[35] A.M. Martinez, The AR Face Database, CVC Technical Report No. 24, June 1998.
[36] W. Gao, B. Cao, S. Shan, X. Chen, D. Zhou, X. Zhang, D. Zhao, The CAS-PEAL large-scale Chinese face database and baseline evaluations, IEEE Trans. Syst. Man Cybern. Part A: Syst. Hum. 38 (1) (2008) 149–161.
[37] X. Liu, X. Tan, S. Chen, Eyes closeness detection using appearance based methods, in: Proceedings of the 7th International Conference on Intelligent Information Processing, 2012, pp. 398–408.
[38] G. Huang, V. Jain, E. Learned-Miller, Unsupervised joint alignment of complex images, in: Proceedings of IEEE International Conference on Computer Vision, 2007, pp. 1–8.

Please cite this article as: F. Song, et al., Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.024

[39] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2005, pp. 886–893.
[40] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.
[41] M. Brown, R. Szeliski, S. Winder, Multi-image matching using multi-scale oriented patches, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, 2005, pp. 510–517.
[42] T. Ojala, M. Pietikainen, T. Maenpaa, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 24 (7) (2002) 971–987.
[43] C. Chang, C. Lin, LIBSVM: a library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (3) (2011) 27:1–27:27.
[44] S. Baluja, H. Rowley, Boosting sex identification performance, Int. J. Comput. Vis. 71 (2007) 111–119.
[45] P. Wang, Q. Ji, Learning discriminant features for multi-view face and eye detection, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, 2005, pp. 373–379.

Fengyi Song received the BSc degree in computer science from Henan University, China, in 2006. In 2009, he received his MSc degree in computer applications from Nanjing University of Aeronautics and Astronautics (NUAA), China. He is currently a PhD student at the Department of Computer Science and Engineering, NUAA. His research interests include face recognition and computer vision.

Xiaoyang Tan received his BSc and MSc degrees in computer applications from Nanjing University of Aeronautics and Astronautics (NUAA) in 1993 and 1996, respectively. Then he worked at NUAA in June 1996 as an assistant lecturer. He received a PhD degree from Department of Computer Science and Technology of Nanjing University, China, in 2005. From September 2006 to October 2007, he worked as a postdoctoral researcher in the LEAR (Learning and Recognition in Vision) team at INRIA Rhone-Alpes in Grenoble, France. His research interests are in face recognition, machine learning, pattern recognition, and computer vision. In these fields, he has authored or coauthored over 20 scientific papers.

Xue Liu received her BSc degree in computer science from Zhejiang University of Technology, China, in 2010. In 2013, she received her MSc degree in computer applications from Nanjing University of Aeronautics and Astronautics (NUAA), China. Her research interests include face recognition and computer vision.

Songcan Chen received his BSc degree in mathematics from Hangzhou University (now merged into Zhejiang University) in 1983. In December 1985, he completed his MSc degree in computer applications at Shanghai Jiaotong University and then worked at the Nanjing University of Aeronautics and Astronautics (NUAA) in January 1986 as an assistant lecturer. There he received a PhD degree, in 1997, in communication and information systems. Since 1998, as a full-time professor, he has been with the Computer Science and Engineering Department at NUAA. His research interests include pattern recognition, machine learning and neural computing. In these fields, he has authored or coauthored over 130 scientific journal papers.

Please cite this article as: F. Song, et al., Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients, Pattern Recognition (2014), http://dx.doi.org/10.1016/j.patcog.2014.03.024