EyeLSD a Robust Approach for Eye Localization and State Detection

Benrachou Djamel Eddine[1] · Filipe Neves dos Santos[2] · Brahim Boulebtateche[1] · Salah Bensaoula[1]

Received: 4 December 2015 / Revised: 18 October 2016 / Accepted: 28 December 2016 / Published online: 31 January 2017

Abstract Improving the safety of public roads and industrial factories requires more reliable and robust computer vision-based approaches for monitoring the eye state (open or closed) of human operators. Getting this information in real time when humans are driving cars or using hazardous machinery will help to prevent accidents and deaths. This paper proposes a new framework called EyeLSD to localize the eyes and detect their states without face detection step. For EyeLSD aims, two novel descriptors are proposed: enhanced Pyramidal Local Binary Pattern Histogram (ePLBPH) and Multi-Three-Patch LBP histogram (Multi-TPLBP). The performance of EyeLSD with ePLBPH and Multi-TPLBP is evaluated and compared against other approaches. For this evaluation three independent and public datasets were used: BioID, CAS-PEAL-R1 and ZJU datasets. The set EyeLSD, ePLBPH and Multi-TPLBP have a greater performance when compared against the state-of-the-art algorithms. The proposed approach is very stable under large range of eye appearances caused by expression, rotation, lighting, head pose, and occlusion.

Keywords Eye localization · Eye state measurement · Image processing · Machine learning

1 Introduction

Many computational vision-based application - ranging from human-computer interfaces (HCI) to road safety systems - require more reliable and robust eye state detection task. In advanced driver assistance systems (ADAS), eye closeness monitoring provides relevant information on driver somnolence and lack of attention. Getting this information in real time when humans are driving cars or using hazardous machinery will prevent accidents and deaths.

Drowsy people often exhibit inherent visual characteristics distinguishable across the face, such as eye states, eye blinking and many other visual features [29]. So, analyzing the eye state is crucial for drivers drowsiness detection [3, 9, 13, 29]. Driver's fatigue strongly correlates with a PERCLOS measure [1]. However, knowing eye localization is determinant to know their state.

Eye localization and eye detection are different problems. Eye detection aims to roughly find the eye in a face image. In contrast, eye localization accurately estimates the center position of the eyes [33, 37].

The key challenge for eye localization and state detection (open or closed) is to find the optimal descriptor (set of features). This descriptor has to be immune to changes of illumination, image noise, scales, and rotations. However, it must preserve relevant information about eye state. Scale and rotation are the main reasons for eye localization failure. This challenges the search of an optimal descriptor.

Using a straightforward image decomposition strategy, such as spatial pyramid method [19], is a step towards a descriptor invariant to resolution changes. G. Mahalingam

✉ Filipe Neves dos Santos
fbsantos@inesctec.pt

Benrachou Djamel Eddine
djamel.benrachou@univ-annaba.org

Brahim Boulebtateche
brahim.boulebtateche@univ-annaba.dz

Salah Bensaoula
bensaoula_salah@yahoo.fr

[1] University Badji Mokhtar Annaba, B.P.12 Annaba, Algeria

[2] INESC TEC FEUP campus, Rua Dr. Roberto Frias,
4200 - 465 Porto, Portugal

and K. Ricanek Jr [25] developed a multi-resolution hierarchy of patch-based feature descriptors for periocular recognition. Their approach combines a hierarchical pyramid-like image and Three-Patch Local Binary Patterns (LBP) [26] feature descriptors (TPLBP) [45]. It can accurately describe periocular features. Also, Turtinen and Pietikinen [41] have used spatial pyramid-like image for coding local texture features. Their work validates this approach for processing arbitrary spatial resolutions of the rigid-textures in challenging conditions. Mäenpää and Pietikäinen [24] and Qian et al. [31] present approaches for rigid scene and texture classification. The classification enhancement is due to their spatial pyramid LBP and multi-resolution LBP approaches.

Inspired by the previous cited studies, in this paper, the pyramidal concept is used to build a new descriptor. Also, based on our earlier works [4, 5], we consider that LBPs have enormous potential for ocular region description and show remarkable results in hard localization conditions. This descriptor, based on LBP and pyramidal concepts, is the core of our proposed approach for eye localization and state detection (EyeLSD). EyeLSD is an extension of our previous works [4, 5]. The EyeLSD contains 3 main stages: the first pre-processes the image reducing noise and enhancing textures, the second extracts feature descriptors in key-points of the input image and then the third focus on the establishment of efficient learning strategy by using a statistical classifiers to interpret the described image patches.

From our test, the EyeLSD detector is more robust than the state-of-the-art methods to a wider range of eye appearance variations caused by occlusion and other image conditions.

The main contributions of this work are:

1. A study about the feasibility of locating eyes and recognizing their states without face detection step and under challenging conditions.
2. A study about the performance of a composition of LBP-based descriptors to: describe local and global texture information of the eye patterns; and, be robust to rotation and scale changes.
3. A novel LBP Pyramid-like descriptor built with multiple LBP variants. It is called enhanced Pyramid LBP histogram (ePLBPH*).
4. A novel growing multi-resolution patch-based LBP features descriptor. It is called growing multi-resolution Three-Patch LBP joint to Gaussian filtering (Multi-TPLBP) and extends the standard TPLBP [45].
5. A novel approach called EyeLSD for eye localization and state detection. EyeLSD can accept several LBP variants including the proposed ePLBPH* and Multi-TPLBP.
6. Benchmark of eye localization accuracy, of EyeLSD considering several configurations, under two public Grand Challenging Face detection databases: BioID (http://www.bioid.com/downloads/software/bioid-face-database.html) and CAS-PEAL-R1 [10]. Benchmark of eye states detection accuracy, of EyeLSD, under real world ZJU eye blink database [32].

This paper is organized with six sections. Section 2 makes a review of the related work. Section 3 presents the proposed framework, EyeLSD for eye localization and its state detection. Section 4 presents the EyeLSD stage to extract the feature descriptor in image key-points. It also presents in detail the two novel descriptors proposed (ePLBPH* and Multi-TPLBP) and its different extensions used in core feature extraction. Section 5 details the experiments realized, including the experimental setup, the databases used, presents the results obtained and compares these results against the state-of-the-art. Finally, Section 6 presents the main conclusions, remarks and future work.

2 Related Work

This section presents a brief review about state-of-the-art approaches for locating the eye and detecting the eye state.

In driver safety domain, González-Ortega et al. [13], developed a real-time vision-based approach that locates eyes and recognizes their states. A hybrid approach is applied for eye state detection, which combines the appearance and the shape of eye features. A projection function is built to distinguish the different eye states (open, nearly closed, and closed). To this end, two interconnected classifiers: support vector machine (SVM) [42] and multilayer perceptron (MLP) [49], formed to interpret each eye state. The algorithm achieved good results under different conditions. However, it may fail under poor imaging conditions (low resolution, blur, and uneven light), that lead to ambiguous appearance of the eye.

Lately, Cui Xu et al. [50], proposed an eye states detection method. They consider the detection of eye states as a binary classification problem, this means that eyes are classified into one of the two categories: closed or open. The eye image is first scanned with a series of scalable sub-windows, where is extracted LBP histograms. Then, for each sub-window, an optimal reference template is trained. Based on reference templates, the bin-wise distances between extracted histograms and the corresponding templates are calculated to build the training feature set. This set and AdaBoost-based classifier are used to locate the eyes and to recognize their states. In theory these feature-descriptors are built to tolerate slight texture rotation. However, they are not invariant to high rotations. Moreover, the length of their LBP feature sets are relatively long, which may increase the computational cost.

More recently, Fengyi Song et al. [38] proposed an eye closeness detection approach in still face images. In their work, face portion is firstly detected and cropped, then enhanced pictorial structural model [39] is adopted to find the eye locations. To define the eye state, they combine local and global structural appearance of eyes to build their state model. This state model uses a multi-scale histogram of principal oriented gradients (MultiHPOG) features. Their algorithm was validated on challenging eye datasets. Hashem Kalbkhani1 et al. [18], proposed an algorithm that estimates open and closed status of eyes in colored images. Their framework consists to crop the facial region first, and then retaining only 60 % of its upper area, which will be pre-processed after. The eyes are detected in this predefined region, using an improved version of Eye-Map algorithm [35]. After that, a fine estimation of the pupil center (iris) is set as the center of mass of the ocular regions. In the last stage, the eye is set as open if the number of white pixels inside the iris circle in a binary sub-space is more than the number of black ones. Otherwise, the eyes are closed. Their algorithm achieves high recognition rate, and does not require training data. However, detecting face step is required as prerequisite for eye localization success. Without this step, the detection of the eyes as well as the recognition of their state fail. Also, their framework is not applicable if two eyes are not completely appeared in case of extreme face rotation for example. Thus, the rotation of the face has serious effect on eye region detection.

From these works we can state that a reliable eye state detection requires high accuracy in eye localization. The eye localization in the real world scenarios is challenged by eye appearance changes due variance in scale, orientation, rotation, head pose, expressions, illumination, arbitrary image resolutions, occlusion, and even the inter-individual variability. These factors pose a great challenges to this task and introduce significant changes to the visual aspect of eye patterns. So, enormous efforts are invested in research to solve these difficulties.

In the literature, eye localization techniques can be divided in four categories [37]: the characteristic-based, the structure-based, the appearance-based, and hybrid-based approaches.

The characteristic-based approaches focus on exploiting distinct inherent characteristics of eyes. Several eye-specific characteristics can be reliably used in practice. For example, the particular eye shape, the intensity response and contrast sensitivity of the eye components. However, these approaches show limitations under uncontrolled nature settings. Traditional methods that measure previous information are Integral Projection Function (IPF) [8] and Integral Variance Function (VPF), both of which are merged in General Projection Function (GPF) [55]. These methods yield a good localization results, but most of them show a high efficiency only with normalized face images (slight change in eye scales and texture rotations). Moreover, they tend to be less efficient under uncontrolled conditions and facing poor image quality, which may result in great performance loss. Li et al. [22] try to overcome a weakness of such methods, by accumulating locally smoothed version of pixel intensity, which tends to be more stable compared to the global one.

The structure-based approaches, place emphasis on the study of the spatial structure of the eye intrinsic components and the geometric regularity between eyes and other facial features in the face context. The Active Shape Model (ASM) is a popular method used to model the structure information of the eye. ASM counts several extensions and is applied in different vision-based applications [7, 17, 34, 46]. The eye structure-based approaches, infer successfully the location of an eye by estimating the locations of its parts, and show robustness against occlusion and large eye variations. Among existing methods, Hough-transform technique, realized good performance for eye locating recently [20, 21]. However, this technique is computational expensive. The enhanced pictorial model [39] achieved also promising performance, and shows robustness to illumination and adaptability to eye localization in unconstrained situations.

The appearance-based approaches build a statistical model of the eyes, from their photometric appearance. The eye patterns are described through their visual appearance by combining extracted features and statistical classifiers. Three steps summarize these approaches; (1) pre-processing (image noise reduction, illumination correction), (2) feature extraction and normalization, (3) classification and building a learning-model, which interprets the extracted features. The advantage of these methods compared to that described before, is that a richer and reliable information about the eye patterns can be obtained, even with low-quality face images [51]. Nevertheless, the appearance-based approaches count some limitations. They can be less optimal for locating eyes and inefficient to capture their large visual changes (variations in scale, rotation and occlusion.), although, many existing studies on localizing eyes, these issues are not completely solved. A fully reliable method does not yet exist. Some of the proposed approaches are application oriented or, they are constrained to several conditions, e.g., near frontal view angles, uniform lighting, and fully open eyes. As well as, eye appears differently at arbitrary spatial image resolutions, and facial region captured at different depth-of-field, with different orientations, rotation and other variations. These form a real challenge for any appearance-based algorithm. Moreover, it is difficult to address all variations by using a single type of feature set. The choice of the descriptor is application-driven and depending on the establishment of searching strategies.

To summarize, most of the aforementioned methods, are capable of giving workable solutions to eye localization under restricted settings. The essence of eye localization is mainly used to help eye detection with finer estimation of eye location, whatever changes that faces or eyes can undergo. Recently, Yan Ren et al. [33] proposed a learning method for precise eye localization. They combine a two-class sparse representation classifier (SRC) and scale invariant feature transform (SIFT) features to keep invariance to arbitrary scale and rotation. The search for an eye location is tackled by creating a heat-map with SRC output and pyramid-like locating method that discriminates the eye from a non-eye under variant resolution, while reducing the amount of searching regions. Their method shows feasible eye localization without the assistance of face detector. Shiming Ge et al. [11] formulates the eye localization as an optimization problem. Their method is based on correlation filter bank (CFB) trained with EM-like adaptive clustering technique. The final model can find the exact eye location under pose and illumination changes. Mingcai Zhou et al. [54], investigate eye localization problem, by using coarse-to-fine searching strategy and improving Supervised Descent Method (SDM), joint to multiple nonlinear features that enhance the accuracy, while maintaining a certain inviance. Their approach is called coarse-to-fine multi-feature SDM (CF-MF-SDM). The CF-MF-SDM algorithm achieves better localization accuracy when compared to other methods, but fails to locate eyes with large rotation angles in-plane.

3 Eye Localization and State Detection Approach (EyeLSD)

To locate the eye and detect its state without face detection step, we propose the EyeLSD framework, Fig 1. EyeLSD combines the strength of several descriptors that properly describe the eye appearance. Their combination provides richer and consistent information of the eye.

EyeLSD is decomposed in three main stages, as shown in Fig. 1. The first stage consists to pre-process the original image by reducing noise and enhancing textures (steps a and b), the second stage extracts features in key-points of the image (steps c,d,e, and f), and then the final stage uses statistical classifiers to interpret collected information (step g).

In stage one, Fig. 1a, the 3-channel RGB image is converted into gray-scale image Υ, then image processing techniques are applied in Υ, to filter out noise and further enhance the localization results.

After a searching map is created, with key-points that highlights different facial traits (e.g., nose tip, mouth corners, eye centers, eyebrow, and lips). This pre-processing

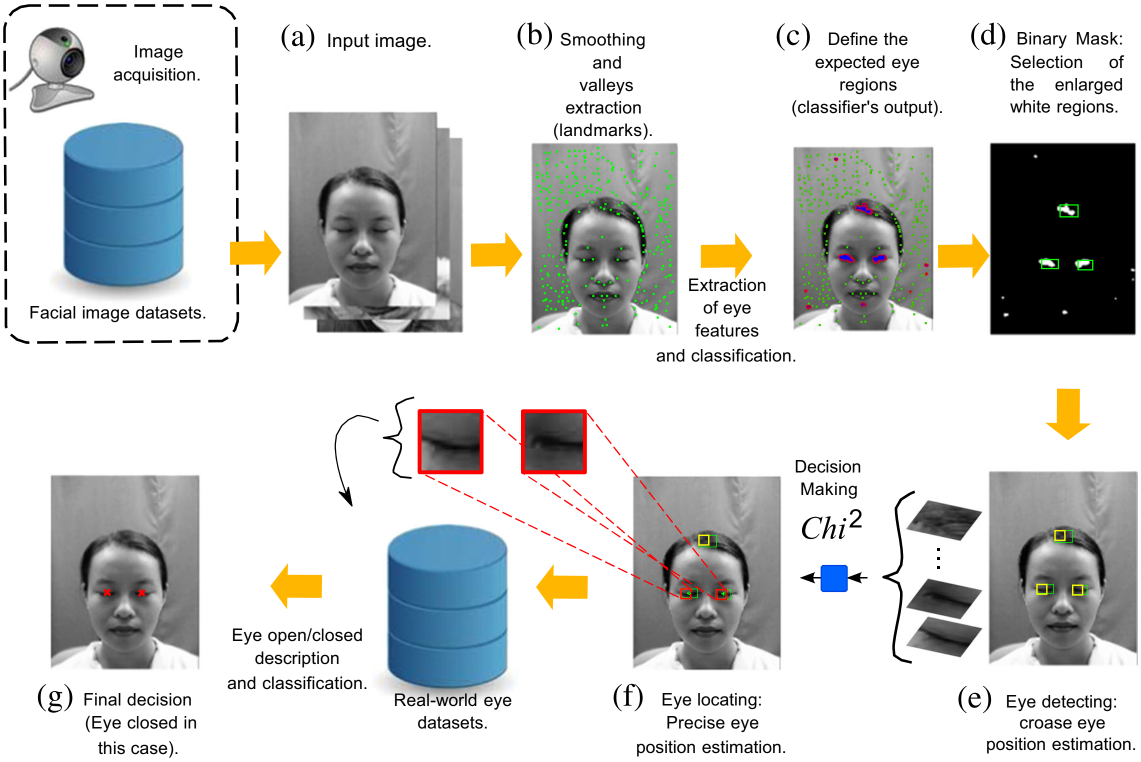


Image acquisition.
Facial image datasets.
(a) Input image.
(b) Smoothing and valleys extraction (landmarks).
Extraction of eye features and classification.
(c) Define the expected eye regions (classifier's output).
(d) Binary Mask: Selection of the enlarged white regions.
Decision Making
Chi^2
(e) Eye detecting: crease eye position estimation.
(f) Eye locating: Precise eye position estimation.
Real-world eye datasets.
Eye open/closed description and classification.
(g) Final decision (Eye closed in this case).

Figure 1 EyeLSD flowchart; eye localization and open/ closed state estimation.

step is used as locating strategy to meet requirement of localization, instead of using traditional sliding-window strategy, that gradually moved over the whole image pixels with a fixed step (relatively small), at risk of missing important image regions, e.g., the ocular region.

In stage one, Fig. 1b, to built the searching map with key-points, the image Υ is firstly blurred because, LBP based descriptors are sensitive to noise in near-uniform image regions. Then, local minimum regions (valleys) are extracted from the pre-processed image, corresponding to the lower gray-scale regions. These regions are the most likely places to contain the true position of facial traits. Connected-component labeling is introduced to detect connected regions in valleys, while assigning them landmarks (key-points) for the next steps.

In stage one, Fig. 1c, the spatially enhanced Pyramid-like method, i.e., ePLBP*, Section 4, is used to encode local features with different scales within key-points. The ePLBP* descriptor reduces the influence of illumination and noise change, while moderates the variation in scales and rotation degrees.

The second stage consists of using the region selection methods, to choose the most discriminated image parts that stand for ocular regions. The spatial structure of objects in a scene is used, besides of binary morphological operations. In the morphology step, there are two phases, erosion and dilation [12]. The white region (pixels) of binary image, are expanded and the black region (pixels) are diminished by erosion operation. Afterward, black region (pixels) of the eye area diminished by the erosion operation is expanded by dilation operation. This second process of sequentially erosion and dilation is called opening process. After the opening process applied two times, the largest blob is generated for an eye (enhanced area) and small blobs are generated for noise then rejected.

In stage two, Fig. 1d, the measure of the structural proprieties of the enhanced area (retained) is applied upon binary image to choose the widest surface, i.e., the silhouette of the eye. The area's center of mass (moment) is considered as an eye center.

In stage two, Fig. 1e, a bounding-box is reported-back on the original input image and detected regions are geometrically normalized to 24 × 24 pixels. The eye candidates are classified as true or false based on similarity value *Chi-square* ($\mathbb{S}_{\chi^2}$), which boosts the performance of the ocular detector. This post classification phase increases the possibility to localize eyes, whatever their states as shown in Fig. 1f.

The final stage consists of detecting the eye open and closed status. This task is introduced as a 2-class problem based on the eye appearance. From the located eyes, various feature sets are extracted to feed the classifier for final decision, as shown in Fig. 1g. The descriptors used are detailed in Section 4.

4 EyeLSD: Feature Vector Extraction and Classification Stages

This section presents two new descriptors to help the two stages of eye localization and eye states detection. The proposed names for the two novel descriptors are: enhanced Pyramidal Local Binary Pattern Histogram (ePLBPH) and Multi-Three-Patch LBP histogram (Multi-TPLBP). The classification performance is enhanced (step e of EyeLSD, Fig. 1), by using a similarity measurement that accurately locates the eye center.

4.1 Descriptor for Ocular Region Detection

This subsection firstly presents a feature descriptor based on Local binary pattern (LBP) and Spatially enhanced local binary pattern histogram (eLBPH). Then, based on this descriptor, we propose a new extended version ePLBPH. Both can locally and globally describe the eye shape, under different conditions. These descriptors can be easily implemented using a MATLAB source code available at http://www.ee.oulu.fi/mvg/page/lbp_matlab.

4.1.1 Feature Descriptor based on Local Binary Pattern (LBP) and Spatially Enhanced Local Binary Pattern Histogram (eLBPH)

The local binary pattern (LBP) is a powerful gray-level invariant texture primitive. The non-parametric LBP operator was firstly mentioned by Harwood et al. [14], and then introduced by Ojala et al. [26] for texture description. The original operator works with a 3 × 3 neighborhood. The pixel values of eight neighbors are thresholded with respect to the center pixel, then, the so-thresholded binary values are weighted by powers of two and summed to give the LBP code of the center pixel.

In practical tasks, the statistic form of LBP codes, LBP histogram (LBPH), generally is used. That is, the LBP codes of all pixels for a given image are collected into a histogram as a texture descriptor.

A simple extension of the LBP, denoted by $\text{LBP}_{P,R}$ is to use neighborhoods of different sizes [28]. The extension can take any radius (R) and neighbors (P) around a center pixel, by using a circular neighborhood and the bilinear interpolation whenever the sampling point does not fall in the center of a pixel.

Another extension is the so-called uniform patterns $LBP^{u2}_{P,R}$. A LBP code is called uniform if it contains at most two bitwise transitions from 0 to 1 and conversely when the bit pattern is moved in a circular binary form [28].

For the computation of LBPH, the uniform patterns are used such that each uniform pattern has an individual bin and all non-uniform patterns are assigned to a separate bin. So, with 8 neighbors, the numbers of bins for LBPH are 256 bins and 59 bins for uniform patterns LBPH ($LBPH^{u2}$), respectively. Clearly, the uniform patterns reduce the length of feature vectors, without a significant information loss.

The ocular region is considered as dynamic and non-rigid object, highly sensitive to imaging conditions and environmental variations. So, by pre-processing the image patch into several sub-region (sub-block), we can mitigate these large variations to a certain extent. The resulting texture descriptor is called enhanced LBP histogram (eLBPH), which is chosen to describe the eye with a $LBPH^{u2}$.

The eLBPH proposed by Ahonen et al. [2], is a reference for LBP based face recognition techniques. The eLBPH implementation for facial area description consists of the following procedure: first divide the facial image into d sub-regions $\{R_0, R_1, \ldots R_{d-1}\}$ and from each sub-region the LBPH is calculated individually, then the resulting d sub-regional LBPHs are concatenated to form the eLBPH, in the same order of the regional division applied to the image. The eLBPH descriptor has a length of $d \times l$, where l is the length of the sub-regional LBPH. Figure 2 shows an illustrative example of the sub-regional division and histogram concatenation strategy for eye representation.

LBP^{u2} is statistically stable and less sensitive to noise [28]. All sub-regional LBPH are concatenated to form the eLBPH of 236 bins (59 bins × 4), Fig. 2. These parameter settings were suggested by [5] for ocular region description. They have shown that the eye is effectively represented by eLBPH in three different forms:

1. The labels of the local histogram contain information about the eye at a pixel-level.
2. The labels are summed over a sub-blocks level.
3. The sub-block histograms are concatenated to build a spatial enhanced description of the eye.

4.1.2 Enhanced Pyramidal Local Binary Patterns

Local binary pattern in spatial pyramid domain (PLBP) is a powerful multi-resolution analysis method. Over the pyramid transformation, each pixel in the low spatial layer of the pyramid, is generated by down sampling the low-pass filtered high resolution image at the pyramidal level just below as shown in Fig. 3. So, in images of low-resolution, a pixel corresponds to a region in its high-resolutions. This region is described by [23, 24] as an *"effective area"* of the filtered pixel. Please see [31] for more details about LBPs representation in spatial pyramid domain. The pyramid generation approach consists of low-pass filtering (LPF) and down sampling images at the pyramid level just below. The pyramid image is recursively constructed as follows:

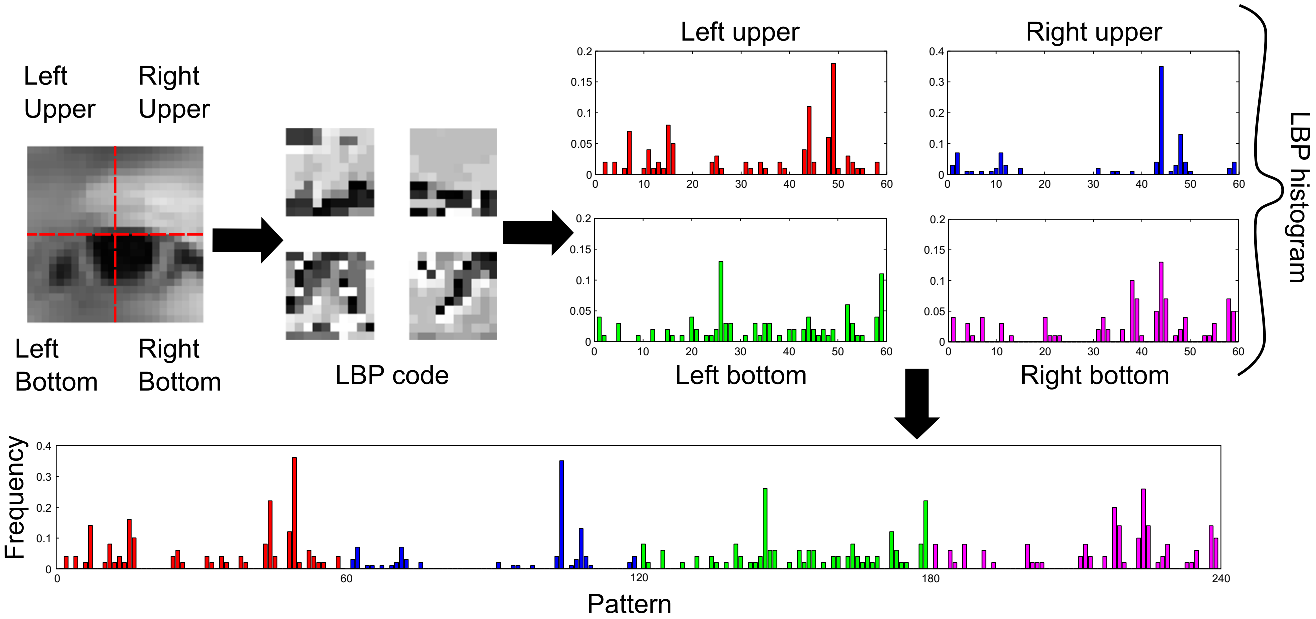


Figure 2 Enhanced LBP histogram (eLBPH). The eLBPH is used to describe the ocular region. It is formed on the basis of an eye image with size 24 × 24 pixels, which is divided into 4 non-overlapped sub-blocks of size 12 × 12 pixels [2].

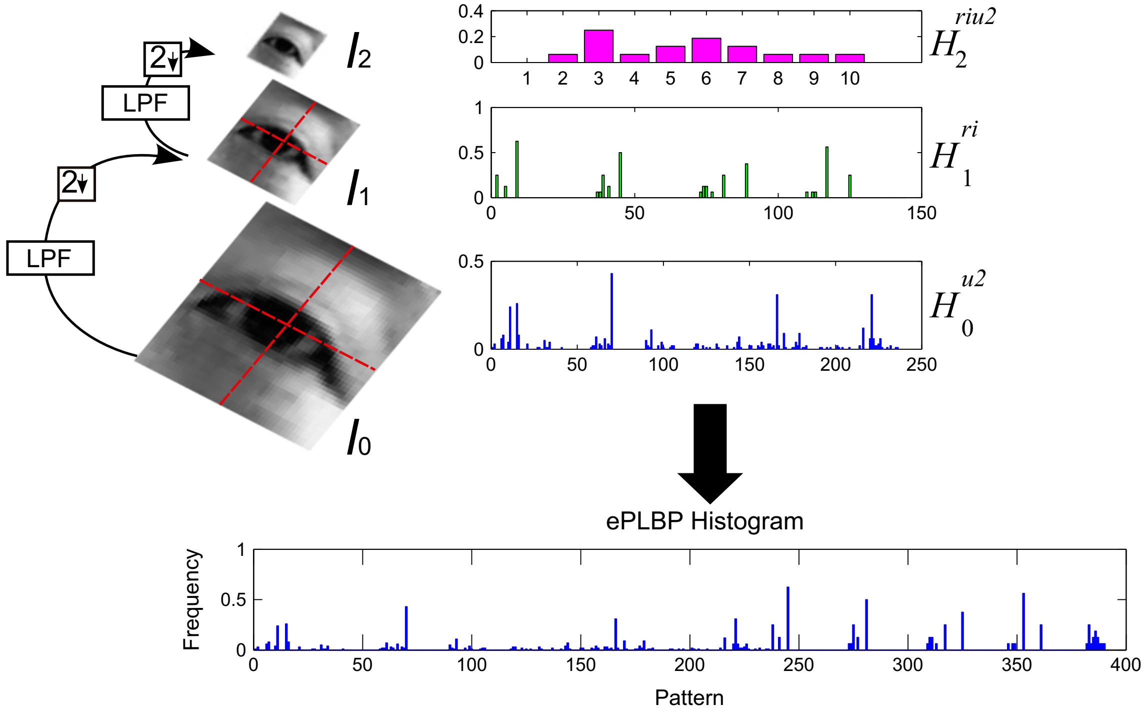


Figure 3 The pyramid decomposition and corresponding LBP signatures. The diagram of pyramid sampling in neighboring 3 resolutions. The down sampling ratios in each x and y directions are both 2, the resolution variation of neighboring two pyramids is with a factor 4.

$G_1(x, y) = I(x, y)$ for pyramid level $l = 1$, where $I(x, y)$ is the original image, $l = \{1, \ldots, L\}$, and L is the number of layers in the pyramid. In a general definition of pyramid construction process and for pyramid level $l > 1$:

$$G_l = \sum_m \sum_n f_G(m, n) G_{l-1}(R_x x + m, R_y y + n) \tag{1}$$

where R_x and R_y are the down sampling ratios in x and y directions, respectively. $(R_x, R_y > 1)$ in case of down sampling is used during the pyramid image generation. Otherwise, $R_x = R_y = 1$ if no spatial sampling is used. x and y are the image coordinates, whose values are expressed in a Cartesian coordinate system, f_G is a 2-D isotropic Gaussian (circularly symmetric).

In the spatial pyramid domain, feature extraction of texture Γ are produced through a combination of texture information of all pyramid levels. Let Γ^k represent the texture information of the kth pyramid, $(k = 1, \ldots, N)$, g $_c^k$ corresponds to the central pixel of the kth pyramid.

$$\begin{aligned}\Gamma^k &= t(g_c^k, g_k^0, \ldots, g_k^{p-1}) \\ &\approx \gamma(s(g_0^k - g_c^k), \ldots, s(g_k^{p-1} - g_c^k))\end{aligned} \tag{2}$$

The resulting binary code is denoted as LBP$_{P,R,k}$, which is the LBP code of a pixel at the kth spatial pyramid and expressed as follows:

$$LBP_{P,R,k} = \sum_{p=0}^{P-1} (s(g_k^p - g_c^k) 2^p) \tag{3}$$

The final PLBP is a combination of several LBP histograms for N-spatial pyramid images:

$$PLBP_{P,R} = \cup_k LBP_{P,R,k} = \{LBP_{P,R,1}; \ldots; LBP_{P,R,N}\} \tag{4}$$

In our enhanced LBP spatial pyramid architecture ePLBP (proposed for EyeLSD), the eye pattern is down-sampled twice. The spatial pyramid is generated with 3 levels of image sequences $I = \{I_0, I_1 \ldots, I_{L-1}\}$, Fig. 3 (i.e., $L = 3$). The pyramid images are denoted as I_0, I_1, and I_2 for the pyramid's basis, first and second level of the pyramid, respectively. The size of the n^{th} level image has the half size of the $(n-1)^{th}$ level image, Fig. 3.

Based on Ojala et al's rules, the enhanced PLBP$^*_{P,R}$ (ePLBPH) can be constructed, where $*$ stands for $\{u2, ri, riu2\}$ patterns.

The original image, represents the 0^{th} level image I_0 of the pyramid, LBP^{u2} is performed upon the 0^{th} pyramid's image with size 24 × 24 pixels. The uniform patterns are used because they tolerate rotation better, since they contain fewer spatial transitions exposed to unwanted changes upon image rotation, besides being highly descriptive. LBP^{u2} is statistically stable and less sensitive to noise [28]. I_0 is the highest resolution image in our pyramid and contains more details about the eye appearance.

To enhance the discriminative capability of the applied descriptor, the 0^{th} image of the pyramid is equally divided into 4 non-overlapped sub-regions. The global spatial histogram is computed by concatenating all sub-regional LBPH^{u2}. The obtained LBPH^{u2} signature is denoted as H_0^{u2} with a length of (59 bins × 4).

At the 1^{st} level of the image pyramid I_1 with size 12 × 12 pixels, LBP^{ri} code [27] is generated, to handle the invariance to texture rotation. This pyramid image level is pre-processed in same way as I_0. The LBP histogram obtained is denoted as H_1^{ri} with a length of 36 bins × 4. The I_1 is a smoothed image of size 12 × 12 pixels and contains less details about the eye appearance. So, the extracted information may not be very discriminant. To solve this issue, we proposed to use a sub-region-division and histogram concatenation strategy. It also improves the discriminative power of the applied descriptors and preserves the spatial relation for the eye. The full eye image is divided into 4 sub-regions, as shown in Fig. 2. The spatial relation of the four parts is basically preserved in the concatenated histogram.

The highest level image I_2 may be sparse and unstable, due to its small size of 6 × 6 pixels, and thus the sub-region division cannot be used therein. At this level of the pyramid, the feature descriptor used should be highly descriptive, while generating a small length of LBPH. The LBP^{riu2} [53] provides a good discrimination in comparison to the "*non-uniform*" patterns, and this leads to differences in their statistical properties [28]. However, in this pyramid image level the spatial preservation of LBPH does not preserve any spatial information of eye, due to its histogram statistic over the whole eye image patch. The LBP^{riu2} is by definition gray-scale invariant measure, ensures the invariance to rotation, and considers only uniform patterns, this is a fundamental proprieties of texture. The LBP^{riu2} generates a histogram denoted as H_2^{riu2} with a length of 10 bins. Finally, all histograms H_0^{u2}, H_1^{ri} and H_2^{riu2} are concatenated, to form the enhanced pyramidal eye signature **F**; $F = \{H_0^{u2}, H_1^{ri}, H_2^{riu2}\}$. After that, for each image patch, we could estimate its corresponding features. In this work, the performance of the ePLBPH has been compared with LBP^{riu2} pyramid histogram (ePLBP^{riu2}) of ((10 bins × 4) + (10 bins × 4) + 10 bins) dimensions, which processed in the same way as ePLBPH*.

4.2 Descriptor for Eye State Classification

A family of patch-based descriptor is adopted to encode further types of micro- and macro-textures of the eye images. The proposed descriptor is a Three-Patch LBP that growing in resolutions, joint to a *Gaussian filtering*. Thus, enhances the discriminative power of the original TPLBP, that basically encodes the similarities between pixels neighboring patches of the image in different resolutions, and hence captures a complementary information to that of pixel-based descriptor.

4.2.1 Feature Description using Growing Multi-Resolution TPLBP Combined with Gaussian Filtering (Multi-TPLBP)

The Multi-TPLBP extends the TPLBP descriptor [45] by calculating over different scales (multi-resolutions) of an image. The TPLBP of a pixel is obtained by comparing the values of three patches to provide a single bit value in the code assigned to the pixel. TPLBP for each pixel is computed by taking a window $\omega \times \omega$ of region centered on the pixel and considering m sampling points in a perimeter of radius r pixels.

The TPLBP takes m patches around m pixels in the neighborhood, distributed uniformly on every side of the center patch. The inter-patch comparison in TPLBP is made by comparing the value of the center patch with a pair of patches that are α patches apart along the neighborhood circle. The value of a unique bit is set according to the similarity $d(.,.)$, of the two patches with the center patch. The function $d(.,.)$ is any similarity distance function between two patches (L_2 *norm* in our case). The resulting code has m bits per pixel and denoted as TPLBP$_{\mathbf{R},m,\omega,\alpha}$. Please refer to [25, 45] for more details about TPLBP operator.

The multi-resolution representation provides robustness to the original TPLBP, by collecting intensity information from a larger area, rather than the original single pixel. However, it might be noise sensitivity as sampling is made at single pixel locations, without pre-processing.

The standard multi-scale mechanism counts some shortcomings and does not describe well the image textures due to following reasons:

1. The sampling is done at a single pixel location, rather than considering the effective region [23, 24].
2. The sparse sampling used by TPLBP in a large perimeter (radius) may not result in an adequate representation of the two-dimensional image signal, which involves an aliasing effect [23, 24].
3. The TPLBP is less stable by increasing the neighborhood radius, due to minimal correlation of the sampling points with the center pixel.

To solve this issues, an exponentially growing multi-resolution (Multi-TPLBP) is built by using low-pass filtering TPLBP (TPLBPF). The sampling positions are joint to a filtering process, to cover the neighborhood as well as possible and minimizing redundant information that may be collected by the operator. The Multi-TPLBP extracts both micro- and macro-structures of the eye pattern, which is a real need for efficient information retrieval and contributes positively to the description of the eye open and closed status.

The eye image is pre-processed with a low pass filter (LPF), hence, the intensity information of a sample can be captured from a large area than the original single pixel, which is drawn with a solid circles in Fig. 4.

In LBPF, the m_n circles are with equal sizes and tangency [23, 24]. m_n circles are ensured to be tangency, if their radius is expressed as:

$$r_n = r_{n-1}\left(\frac{2}{1 - sin(\pi/m_n)} - 1\right), n \in \{2, \ldots, N\} \qquad (5)$$

where N is the number of scales and m_n is the number of neighborhood samples at scale n. The low-pass filtering is useful only with radius larger than one for $m_n = 8$, r_1 is set to 1.5, which is the shortest distance between the center and the border of a 3×3 window.

The choice of the TPLBP radius is not randomly made, but according to the rule that the effective areas touch each other [24, 31], Fig. 4a. Therefore, the neighborhood radius at scale $n(n \geq 2)$ illustrated with dashed circles in the Fig. 4a is determined as follows:

$$\mathbf{R}_n = \frac{r_n + r_{n-1}}{2} \qquad (6)$$

The effective area is realized with LPFs designed, so, that 95 % of their mass lies within the solid circle [24, 31], Fig. 4a.

The spatial size (width and height) of the Gaussian filter at scale n is calculated as follow:

$$w_n = 2\lceil \frac{r_n - r_{n-1}}{2} \rceil + 1 \qquad (7)$$

where Eq. 7 is the symmetric weighting function with $(2K + 1)$ taps, that gives the Gaussian kernel f_G in terms of the rules of separable and symmetric. In this case Eq. 7 approximates the Gaussian function. Therefore, Eq. 1 can be reformulated as follows:

$$G_l = \sum_{m=-K}^{K} \sum_{n=-K}^{K} w(m, n) G_{l-1}(R_x x + m, R_y y + n) \qquad (8)$$

where $R_x = R_y = 1$, which means no down-sampling used during the multi-scale image generation. The standard deviation of the Gaussian filter at scale n can be calculated from

$$\sigma_n = \frac{r_n - r_{n-1}}{2\sqrt{-2ln(1-\rho)}}, \rho \in [0, 1] \qquad (9)$$

The effective areas in Multi-TPLBP, are realized with LPFs, where ρ in Eq. 9 is the probability that the mass of the distribution lies inside the solid circles of radius r. (usually, ρ is set to 0.95)

To summarize, the procedure used for building the Multi-TPLBP is very similar to that used by [28]. The only difference lies with the neighborhood samples with radii greater than one are obtained via low-pass filtering. Furthermore, neighborhood radii are chosen following the rules presented before. The final Multi-TPLBP signature is obtained by concatenating the extracted TPLBP histograms at each

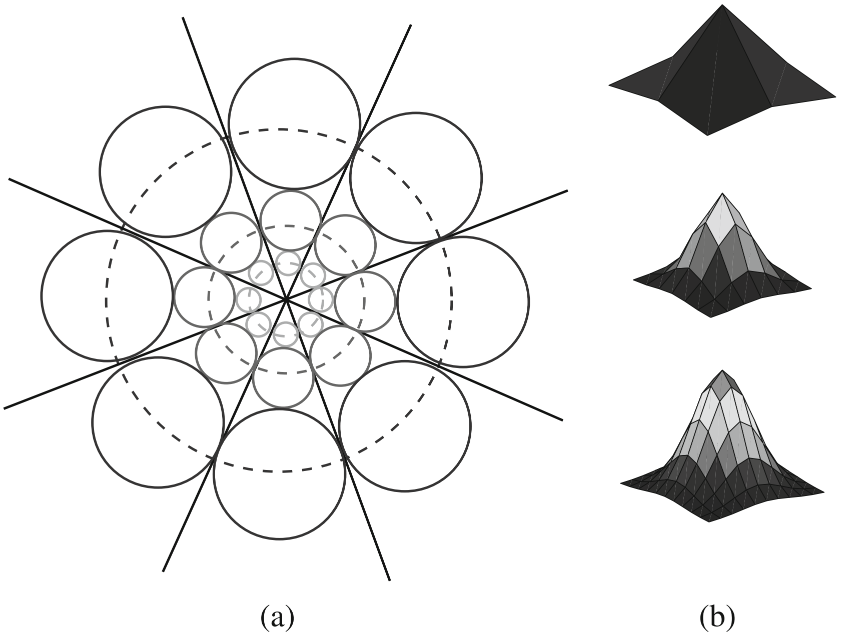


Figure 4 **a** The effective areas of TPLBPF and LBPF of filtered eye images in an $8 - bit$ multi-resolution LBP operator. The *dashed circles* are the radius of the TPLBP rings. Sampling points P_n equally spaced circles with radius r_n (5) and centered on the dashed circles with a radius $\mathbf{R}_n$, which are related to the effective region of each the image pixels. **b** Different Gaussian filter resolutions that can be used in the $1st$, $2nd$ and $3rd$ scales of the image [24, 31].

scale. The Multi-TPLBP maps the eye image into $\mathbb{R}^{N \times d}$ representation, where d is the length of a single TPLBP code histogram at a scale n.

4.3 Similarity Measurement for Precise Eye localization

In this section similarity measure is introduced as a means of boosting the performance of our ocular detector. The problem of estimating the similarity distance between two histograms is quite relevant in several vision-based applications. As a distance measure on the eye localization efficiency, we use the $\mathbb{S}_{\chi^2}$ distance for matching similarity between a pair of histograms that supposed representing the eyes in different states, i.e., given a detected eye patch I' and a template of reference T, which are encoded using LBP descriptor, the $LBP_{16,2}^{riu2}$ histogram is used in our case. The pairs are considered to match if $d(LBP(I'), T) < \tau$, where d is a distance function and τ is threshold (set by user through experiments).

$$\mathbb{S}_{\chi^2}(H, T) = \sum_{i=1}^{n} \frac{(H(i) - T(i))^2}{H(i) + T(i)} \tag{10}$$

where H and T are defined as discrete sample and model distributions, respectively. They correspond to the probability of bin i in the sample and model distributions. n is the number of bins in the distributions, $n = 18$ bins in our case. A small corpus of left and right eye images, were used with a size of 24×24 pixels for each eye state (open or closed), which are selected randomly among different people in the databases. These templates can separate the 2-class using $\mathbb{S}_{\chi^2}$ distance.

Our objective is to detect eyes whatever the state, and reject the most probable non-eye according to τ value. It should be point out, that $\mathbb{S}_{\chi^2}$ values vary greatly under hard imaging conditions (e.g, variability in terms of scale change and uneven light). This may lead to ambiguous eye state detection. One possible explanation is LBP features, give a detailed account of the appearance of eye regions while being invariant to rotation and insensitive to the lighting changes. However, it is difficult to make a decision on whether eyes are open or closed with a local representation of the eye appearance by using $LBP_{16,2}^{riu2}$. Nevertheless, we aim to detect the precise eye position, while rejecting images that do not sufficiently represent the eye appearance. Also, the right eye state detection problem, is addressed by our eye state detector in the next step.

5 Experimental Results

This section evaluates the performance of EyeLSD framework, ePLBP* and Multi-TPLBP descriptors to: localize the eye center, and recognize its state (open and closed). It should be remembered that EyeLSD does not require a face detection step. Three datasets were used for this evaluation BioID (http://www.bioid.com/downloads/software/bioid-face-database.html), CAS-PEAL [10], and ZJU [48].

To prove the EyeLSD generalization capability, a new dataset was acquired with people from our laboratory with a normal laptop web-camera. This allows to assess generalization capability of the detection and localization approach in an unseen images/people (images of people not present in the training datasets). Five experiment sets were realized to validate the performance of ePLBPH* and Multi-TPLBP against other descriptors. Table 1 summarizes the conducted experiments, classifiers employed and type of application. LTP and the Gabor descriptors were implemented according [38], to make a better benchmark of our proposed descriptors against the state-of-the-art.

5.1 Classifiers and Parameters Settings

Two state-of-the-art classifiers were selected, SVM [42] and MLP [49]. They were selected because their implementations are open-source, modular, accessible and computationally efficient.

In EyeLSD, the eye location and state detection, are formulated as a 2-class classification problem. In this paper, we use linear and nonlinear kernels for SVM to find the hyper-plane that maximizes the separation gap between the two classes, while minimizing the number of errors for the training set. The hyper-parameters C and σ of the radial basis function (RBF) kernel are optimized by using grid search technique and the cross-validation group (CV). The best values over the CV group were used to build the learning model. The SVM implementation is done with LIBSVM 3.18 [6]. For the MLPs, to find the best neural network topology or architecture, we have trained and tested several configurations, where the number of

Table 1 Table of EyeLSD configurations used (descriptors and classifiers) for the realized experiment sets.

Descriptor	Classifier	Application
LBPH^{u2} [26]		Eye localization
Gabor [30, 52]	SVM (linear) [42]	
LTP [40]		
eLBPH^{u2} [4, 5]	SVM (polynomial) [42]	
eLBPH^{riu2}		
ePLBPH^{riu2}	SVM (RBF) [42]	
TPLBP [45]	MLP [49]	Eye state detection
Multi-LBP^{u2} [24]		
Multi-LBP^{riu2}		

hidden neurons was changed according to feature descriptor size. We keep the number of maximum iterations constant of 10000 and the mean squared error (MSE). Sigmoid functions were selected as transfer functions. The MLP classifier is fully connected and designed with a number of input neurons, equal to the length of each descriptor, i.e., 1440, 3776, 59, 40,90,236 and 390 input neurons for Gabor, LTP, LBPH^{u2}, eLBPH^{riu2}, ePLBPH^{riu2}, eLBPH^{u2} and ePLBPH*, respectively. For the eye detector assessed on BioID dataset (http://www.bioid.com/downloads/software/bioid-face-database.html) and CAS-PEAL-R1 dataset [10], we used 1-hidden layer for each of Gabor, LTP, LBPH^{u2}, eLBPH^{riu2}, ePLBPH^{riu2}, eLBPH^{u2} and ePLBPH*, embedded with 120, 200, 20, 15, 15, 50, and 22 hidden units, respectively.

Regarding, the neural configurations used to build the eye state models by using ZJU eyeblink image gallery [32], each neural model has 1-hidden layer with 12, 25, 100, and 160 hidden units, for Multi-LBP^{riu2}, Multi-LBP^{u2}, TPLBP, and Multi-TPLBP, respectively.

The output vector for positive samples is $\mathbf{Y}_i = (1, 0)^T$ and output vector for negative samples is $\mathbf{Y}_i = (0, 1)^T$. The *Softmax* function is employed in the output layer and the number of output units is equally related to the number of classes. Regularization terms are used to improve the convergence velocity and avoid to settle down in an over-fitting problem. The connecting weights ω are randomly initialized in a range of $(-0.1, 0.1)$, momentum and learning rate are assigned as α and ξ, respectively. During the experiments, the exposed neural architectures are optimized and were found to be a good compromise between MSE minimization and architecture complexity. These neural structures provide a good generalization performance for new data and are valid for the various feature sets.

5.2 Dataset Descriptions

To analyze the eye localization accuracy of our proposed approach, two datasets were used:

- BioID Face database (http://www.bioid.com/download/software/bioid-face-database.html), the BioID dataset contains images in real scenarios with a various illumination, background, and face sizes with and without accessories. The dataset contains 1521 images (384 × 286 pixels, gray level).
- CAS-PEAL-R1 database [10], contains 30900 images (360 × 480 pixels, gray level). This dataset has images acquired in realistic conditions. In our experiments, 1521 face images are selected, 464 subjects with open eyes: frontal slight rotated view; normal and expressions. 330 subjects with accessories, 101 subjects in different background, 302 subjects with eye closed and 324 subjects in different distances from the camera.

We have built datasets composed of images from BioID and CAS-PEAL-R1 datasets, each of which has 3042 eye images and 3419 non-eye images that are manually cropped. 4523 eye and not-eye images are rearranged into a training subset, and the remainder 1938 images with positive and negative samples are confounded and equally divided between test and validation subsets. These images are geometrically normalized into a size of 24 × 24 pixels.

To analyze the eye state detection accuracy of our proposed approaches, a dataset was selected:

- ZJU Eyeblink dataset [32], contains 80 video clips in the blinking video record of 20 individuals, four clips per individual, one clip in frontal view without glasses, one clip with frontal view and wearing myopia glasses, one clip in frontal view and black frame glasses, and

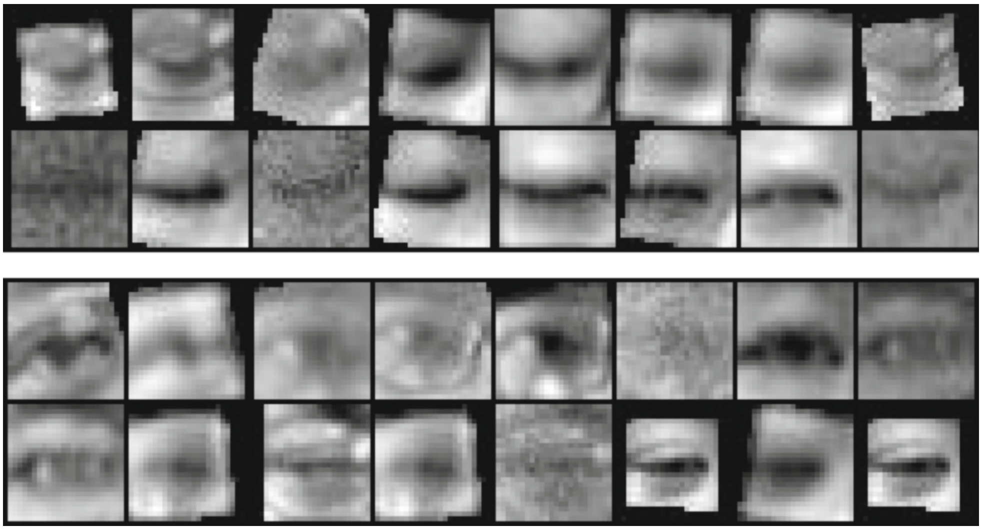


Figure 5 The pre-processed ZJU eye open and closed image gallery: patches in the *top row* are images of closed eyes, and patches in the *bottom row* are images of open eyes.

the last clip with an upward view without glasses. The used dataset in our experiments is rearranged by [38], which expands the variety of the image samples, by adding various transformations, such as rotation, blurring, contrast, and Gaussian white noise.

The ZJU Eyeblink dataset [38], contains in the training subset 1574 closed eyes and 5770 open eyes, and in the testing subset 410 closed eyes and 1230 open eyes. Illustration of eye open and closed images in this dataset can be seen in Fig. 5.

5.2.1 Models Assessment Protocol

The performance validation of the detector and the final results are presented, including calculation of *True Positive* (TP), *False Positive* (FP), *True Negative* (TN) and *False Negative* (FN). The final accuracy *(Acc)* for each descriptor is reported in Tables 2, 3 and 5. The measures used to assess the quality of the learning algorithms are the Receiver Operating characteristic (*ROC*) curves and the Area Under Curve (*AUC*), measured to show the probability of correct discrimination between different classes. Figure 6 shows the *AUC* of each approach's *ROC* graph tested on used databases. *AUC* is calculated using the *10-fold cross* validation technique. *Precision* (Prec) and *recall* (Rec) are computed. Thanks to these two metrics, we compute the *F-Score* interpreted as a harmonic mean of the precision and recall for further comparison of the results. If the trained classifier is predicting a positive class, we mostly have a high Rec and a low Prec, otherwise, if the trained model predicts a negative class, a high Prec against a very low Rec. Thus, model achieves the highest Prec and Rec simultaneously is often desirable.

Table 2 Eye detection: statistical results on BioID database.

Method	TP (%)	FP (%)	TN (%)	FN (%)	Prec	Rec	$F_1 Score$	Acc (%)	AUC (%)
SVM(Linear)									
LTP	45.25	3.04	49.84	1.86	0,93	0.96	0.95	95.10	99.00
LBPH^{u2}	45.20	11.30	41.60	1.90	0.80	0.96	0.87	86.79	94.00
Gabor	44.37	2.94	49.95	2.73	0.94	0.94	0.94	94.32	98.26
ePLBPH*	45.49	2.62	50.28	1.59	0.95	0.97	0.95	95.78	99.09
eLBPH^{u2}	45.73	3.79	49.10	1.36	0.92	0.97	0.95	94.84	98.91
eLBPH^{riu2}	39.02	9.61	43.29	8.06	0.80	0.83	0.81	82.32	90.53
ePLBPH^{riu2}	41.42	6.14	46.76	5.68	0.87	0.88	0.87	88.18	94.95
SVM(Poly)									
Gabor	43.91	4.54	48.35	3.20	0.91	0.93	0.92	92.26	97.00
ePLBPH*	27.39	1.45	51.40	19.74	0.95	0.58	0.72	78.80	95.81
eLBPH^{u2}	7.17	0.42	52.53	39.86	0.94	0.15	0.26	59.71	94.91
eLBPH^{riu2}	30.48	11.25	41.65	16.55	0.73	0.65	0.68	72.14	81.42
ePLBPH^{riu2}	39.35	10.18	42.73	7.74	0.79	0.83	0.81	82.08	90.03
SVM(RBF)									
LTP	45.30	2.89	50.00	1.80	0.94	0.96	0.95	95.30	99.00
LBPH^{u2}	43.75	3.82	49.07	3.35	0.92	0.92	0.92	92.83	98.00
Gabor	42.10	0.31	52.58	5.00	0.99	0.89	0.94	94.69	99.00
ePLBPH*	45.59	1.78	51.12	1.50	0.96	0.97	0.96	96.72	99.58
eLBPH^{u2}	46.15	1.59	51.31	0.93	0.96	0.98	0.97	97.47	99.81
eLBPH^{riu2}	44.69	8.25	44.69	7.97	0.84	0.85	0.85	83.77	91.84
ePLBPH^{riu2}	43.43	4.45	48.5	3.66	0.90	0.92	0.91	91.93	97.76
MLP									
LTP	45.61	1.18	51.75	1.44	0.97	0.97	0.97	97.37	99.00
LBPH^{u2}	44.27	3.50	49.43	2.78	0.92	0.94	0.93	93.70	98.00
Gabor	46.18	1.23	51.70	0.98	0.97	0.98	0.97	97.78	99.00
ePLBPH*	46.38	1.12	51.73	0.75	0.97	0.98	0.98	98.12	99.06
eLBPH^{u2}	45,87	1.17	51.78	1.17	0.97	0.97	0.97	97.65	99.31
eLBPH^{riu2}	41.42	7.08	45.87	5.63	0.85	0.88	0.86	87.29	93.33
ePLBPH^{riu2}	n	4.27	48.64	1.92	0.91	0.96	0.93	93.81	97.64

Table 3 Eye detection: statistical results on CAS-PEAL database.

Method	TP (%)	FP (%)	TN (%)	FN (%)	Prec	Rec	$F_1 Score$	Acc (%)	AUC (%)
SVM(Linear)									
LTP	44.53	3.61	42.27	2.57	0.92	0.94	0.93	96.49	98.49
LBPH^{u2}	42.67	21.31	31.58	4.43	0.67	0.91	0.77	74.25	79.03
Gabor	45.51	2.47	50.41	1.60	0.95	0.96	0.96	95.92	99.20
ePLBPH*	45.26	2.20	50.70	1.82	0.95	0.96	0.96	95.97	99.19
eLBPH^{u2}	44.55	5.15	47.74	2.53	0.90	0.95	0.92	92.31	99.51
eLBPH^{riu2}	34.00	15.71	37.24	13.04	0.68	0.72	0.70	71.24	78.60
ePLBPH^{riu2}	43.62	6.29	46.62	3.47	0.87	0.93	0.90	90.24	96.94
SVM(Poly)									
Gabor	44.42	3.15	49.74	2.68	0.93	0.94	0.94	94.17	98.76
ePLBPH*	17.68	1.82	51.07	29.31	0.91	0.38	0.53	68.76	91.44
eLBPH^{u2}	0.93	0.79	52.11	46.15	0.54	0.019	0.04	53.05	83.20
eLBPH^{riu2}	21.95	13.27	39.63	25.14	0.62	0.47	0.53	61.58	71.27
ePLBPH^{riu2}	42.26	12.43	40.48	4.83	0.77	0.9	0.83	82.74	90.96
SVM(RBF)									
LTP	45.66	2.11	50.77	1.44	0.95	0.96	0.96	96.44	99.51
LBPH^{u2}	42.05	14.18	38.69	5.05	0.74	0.89	0.81	80.75	88.92
Gabor	42.20	0.41	52.47	4.90	0.99	0.89	0.94	94.68	99.56
ePLBPH*	46.06	1.17	51.73	1.03	0.98	0.98	0.98	97.80	99.70
eLBPH^{u2}	45.82	2.48	50.42	1,26	0,95	0,97	0,96	96.25	99.39
eLBPH^{riu2}	39.63	9.95	42.96	7.45	0.80	0.84	0.82	82.60	91.17
ePLBPH^{riu2}	44.28	2.86	50.09	2.76	0.94	0.94	0.94	94.37	98.53
MLP									
LTP	45.04	2.22	50.72	2.01	0.95	0.96	0.95	96.65	97.65
LBPH^{u2}	42.31	5.46	47.47	4.74	0.88	0.89	0.89	89.78	95.55
Gabor	46.38	0.82	52.06	0.72	0.98	0.98	0.98	98.45	99.37
ePLBPH*	45.77	1.31	51.64	1.26	0.97	0.97	0.97	97.42	97.97
eLBPH^{u2}	44.93	2.48	50.46	2.11	0.95	0.95	0.95	95.40	99.39
eLBPH^{riu2}	38.98	8.2	44.74	8.06	0.83	0.83	0.83	83.72	90.65
ePLBPH^{riu2}	42.26	2.62	50.28	1.83	0.95	0.96	0.95	95.54	98.11

5.3 Experimental Results and Discussions

This section presents a performance comparison between different eye descriptors, Tables 2–3. The objective is to show that the enhanced pyramid LBP with high number of bins improves the discriminative power for eye representation. The first experiment 5.3.1 compares performance between eLBPH^{riu2} and ePLBPH^{riu2} to classify the eye presence. The second experiment 5.3.2 compares the performance between eLBPH^{u2} and the ePLBPH*. In the third experiment 5.3.3 the categorization capability of the ePLBPH^{riu2} is compared against the proposed ePLBPH*. The last experiment 5.3.4 analyzes the benefits of ePLBPH* by comparing its performance against those realized by LBPH^{u2}, LTP, and Gabor feature descriptors.

5.3.1 Experiment # 1: Comparison Between eLBPH^{riu2} and ePLBPH^{riu2}

This experiment intends to highlight performance gained if more than a single resolution of descriptors are used in the pyramid image generation. Table 2 shows the descriptors performance on BioID dataset. The best classification accuracies generated by eLBPH^{riu2} are 83.78 % and 87.29 % for SVM(RBF) and MLP classifiers, respectively. By adapting an extension of LBP^{riu2} in a pyramid transform domain(ePLBPH^{riu2}), the approach realized improvements of 8.16 % and 6.52 % with SVM(RBF) and MLP, respectively. Table 3 shows the performance of ePLBPH^{riu2} and eLBPH^{riu2} assessed on CAS-PEAL-R1 image set. The best classification accuracy of eLBPH^{riu2} are 82.6 % and

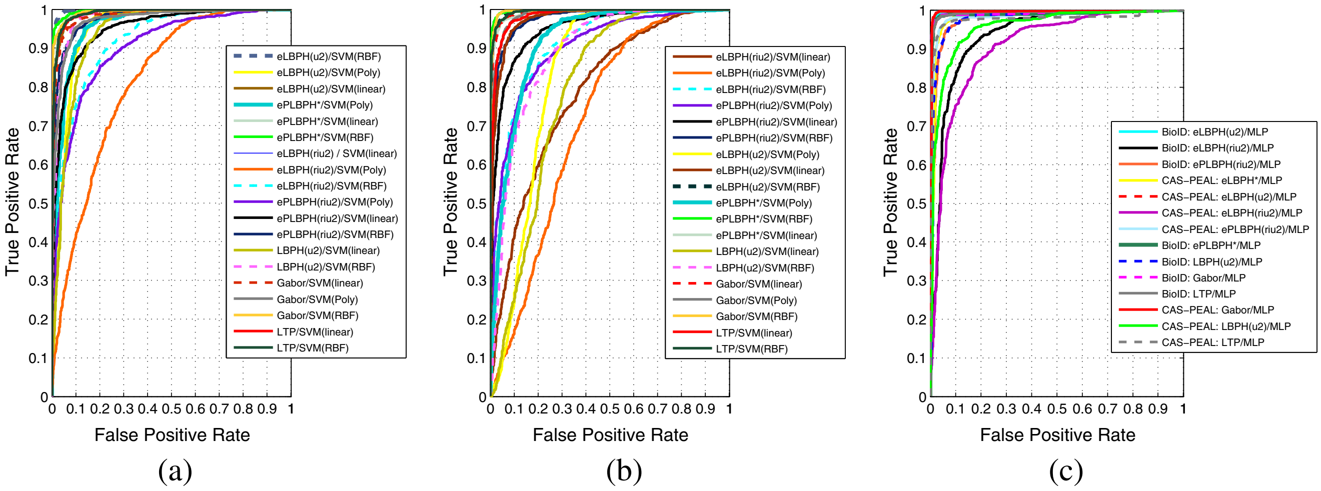


Figure 6 ROC curves of various features using the SVM classifier; **a** BioID database, **b** CAS-PEAL database. ROC curves of various features using the MLP classifier; **c** BioID and CAS-PEAL databases.

83.72 % with SVM solver(RBF) and MLP, respectively. The ePLBPH^{riu2} realized improvements of 11.77 % and 11.82 % with SVM(RBF) and MLP classifiers, respectively.

From the Tables 2 and 3, we conclude that the performance gain of ePLBPH is better than those obtained with eLBPH, by considering descriptors uniquely based on LBPH^{riu2}. The number of ePLBPH^{riu2} pyramid levels is set to 3. The histogram dimension of ePLBPH^{riu2} is (10 bins × 4) + (10 bins × 4) + 10 bins = 90 bins, that is more than 2 times of the eLBPH^{riu2} of size 40 bins (10 bins × 4). The ePLBPH accounts the eye in 3 different resolutions, which increase the information content extracted and the classification rate. Although, the eLBPH describes micro- (edges, corners, spots, etc) and macro-textures (global shape) of the eye pattern, but only at a single resolution.

5.3.2 Experiment # 2: This Experiment Compares Discriminative Performance of eLBPH^{u2} and ePLBPH(Proposed)*

Table 2 shows the performance comparison of eLBPH^{u2} and ePLBPH* on BioID dataset. The ePLBPH* is built according to the Eq. 4, ePLBPH* = {eLBPH*, eLBPH*, LBP*}, $* \in \{u2, ri, riu2\}$. The average performance assessment of eLBPH^{u2} features, yields a best scores of 97.46 % with SVM(RBF) and 97.65 % with MLP classifiers. The performance gained by adopting several LBP variants in spatial pyramid domain (ePLBPH*) are 0.469 % with MLP classifier. Table 3 shows the statistical assessment of eLBPH^{u2} and ePLBPH* on CAS-PEAL-R1 dataset. The eLBPH^{u2} achieves a best scores of 96.24 % and 95.40 % with the SVM(RBF)and MLP, respectively. Among these two configurations, a performance improvement of ePLBPH* over eLBPH^{u2} are 1.54 % and 2.01 %, with SVM(RBF) and MLP, respectively. As shown in Tables 2 and 3, the ePLBPH* performance increases comparing to those of eLBPH^{u2}. So, there shows that the major discriminant properties of ePLBPH* are got from the image into the pyramid basis. This level of the pyramid is pre-processed in the same way as eLBPH^{u2} descriptor. It is noteworthy that applying region division method to form eLBPH is somewhat arbitrary. The division approach spatially enhances the LBP histogram, but also causes both aliasing effect due to the direct sampling and loss of resolution information. The ePLBPH solves these drawbacks, by applying the LPF on the images before pre-processing and LBP histogram calculation, in the 0^{th} and the 1^{st} pyramid image levels. Results show that the proposed method (ePLBPH) achieves a good generalization performance on unseen image set. Hence, the LBP features of a 3-level image pyramid are efficient.

*5.3.3 Experiment # 3: Comparison of the Pyramid Descriptor Performance, Between ePLBPH^{riu2} and ePLBPH**

This experience intends to verify whether performance is further improved, in case of more than a unique LBP mapping scheme are used in ePLBPH. Tables 2 and 3 show the performance comparison of ePLBPH^{riu2} and ePLBPH*, both are realized with 3-level of image pyramid. Table 2 shows improvement realized by ePLBPH* over those of ePLBPH^{riu2} on BioID dataset, which are about 4.78 % and 4.31 % by using SVM(RBF) and MLP, respectively. In Table 3 the classification performance of ePLBPH^{riu2} assessed on CAS-PEAL-R1 dataset, are enhanced by ePLBPH* about 1.87 % and 1.875 % with SVM(RBF)and MLP, respectively.

The ePLBPH* feature vector length is (59 bins × 4) + (36 bins × 4) + 10 bins = 390 bins, that is more than 4 times of the ePLBP^{riu2} histogram of (10 bins × 4) + (10 bins × 4) + 10 bins = 90 bins. The multi-mapping pyramid image ePLBPH*, achieves a higher enhancement and outperforms the three descriptors of comparison; eLBPH^{riu2}, eLBPH^{u2} and ePLBPH^{riu2}. This improvement of the pyramid-like image decomposition with three LBP variants, is statistically significant and observed through the present and the precedent experiment. The ePLBPH* feature sets compensate the information losses during the spatial down-sampling. Also, down-sampling process does not affect the discriminative performance of the descriptor very much. Even so, the enhanced multiple mapping scheme improves significantly the discriminative power of the descriptor, by comparing to that of ePLBPH^{riu2}. This is shown through the experience #1 5.3.1.

5.3.4 Experiment # 4: Comparison Performance ePLBPH* Among Gabor Wavelets, LTP and LBPH(LBPH^{u2}) feature sets

We extend the performance evaluation of ePLBPH*, by introducing a series of comparison with other feature sets, that describe the local shape, the global shape, and the local texture information under difficult conditions. Current feature sets offer quite good performance under illumination variations and many other variations of the real world.

Tables 2 and 3 give the results of LBPH, LTP and Gabor feature sets on BioID and CAS-PEAL-R1 datasets. Please note that the SVM (polynomial) tends to overfit by using LBPH and LTP. For this reason we omitted these results.

In BioID dataset, we can observe that LTP realized a best accuracies of 95.3 % and 97.37 % with SVM(RBF) and MLP with highest AUC value, while Gabor has a slightly worst accuracy of 94.69 % and a competitive performance of 97.78 % with the same settings (dataset and classifiers). The same observations are valid with experiments conducted on CAS-PEAL-R1 dataset.

In this study, we reproduced the LTP code implementation realized by [38]. The threshold value of the LTP code is set to 5 computed from eye and non-eye images of size 24 × 24 pixels, which are divided into 3 × 6 sub-region and each sub-region is represented in an LTP histogram of 59 bins. The resulting LTP feature vector has a 3776-dimensional (32 × 59 × 2). The LTP feature set improves generalization of LBP features, and has a good discriminative capability, while being tolerant to lighting changes and less sensitive to noise in uniform regions.

Gabor filter bank realized competitive results with those of the ePLBPH*, by using polynomial kernel of the SVMs, Gabor outperforms ePLBPH* in terms of classification accuracy. MLP performs well with Gabor parameterization. We observe that in CAS-PEAL-R1 dataset, MLP classifier performs better than SVM(RBF), where 98.45 % of correct classification was obtained for Gabor parameterization, whereas for BioID, Gabor features achieve 96.72 % and 97.78 % with the SVM(RBF) and MLP network classifier, respectively.

This classification enhancement of Gabor features, can be made clear by that CAS-PEAL-R1 dataset contains variation in illumination and pose, but not blurred images, which enhances significantly classification results, since Gabor is a powerful feature descriptor especially for non-rigid texture such face [3] and eyes, whereas BioID databset presents addition variations than previously mentioned, images are blurred. This parameter decreases the efficiency of Gabor method in BioID dataset, but not of our method. The proposed ePLBPH* handles well resolution variations and blurred images. This is proven by the best accuracy achieved in BioID dataset of 98.12 % with MPL network.

Gabor features is implemented with 40 filters [3, 38] (8 orientations and 5 scales) applied on 24 × 24 pixels eye and non-eye patches, then down-sampling the resulting vector by 16. So, instead of (5 × 8 × 24 × 24) that yields 23040-dimensional, it is reduced to a 1440-dimensional vector.

The LBPH^{u2} obtains the lowest accuracies of 93.7 % and 89.78 % with MLP classifier and AUC values of 98 and 95.55 on BioID and CAS-PEAL-R1 datasets. LBPH is not quite appropriate for ocular region description, LBP features are more effective when the eye patterns are pre-processed upstream (sub-region division method). LBPH is sensitive to noise and can slightly tolerate texture rotation but not invariant to that. So, a holistic representation of LBPs can not preserve image local structures in presence of noise, blur, and extreme rotation because the small pixel differences of the eye patterns, make the descriptor vulnerable to noise. Even by using LBP uniform to describe the eyes, we can limit the effect of noise, because most of image local structures can be represented by uniform codes and noise patterns fall into the non-uniform codes. LBPH can only capture small appearance details of the eye patterns.

From precedent experiences, it can be noticed that Gabor, LTP and eLBPH^{u2} feature sets achieved a better performance than those obtained with eLBPH^{riu2} and ePLBPH^{riu2}. The eLBPH^{u2} descriptor is compactly represented in a histogram signature of 236 bins. The eLBPH feature vector requires a sub-region division method as pre-processing stage. It can be observed that effectiveness of eLBPH is not limited on the adopted image decomposition strategy, but related also to the LBP variant used to construct descriptor.

We observe that the LBPH is unfavorable for representing the eye, due to large variations of eye pattern. By contrast, if the whole eye image is divided into sub-regions and the LBPH is individually calculated in each region, as shown in Fig. 2, then variations within-region are less than ones of the whole eye image, in other terms, textures within-region relatively tend to uniform, and thus each regional histogram becomes less fluctuant; with the divided regions, each regional histogram progressively becomes more stable. Hence, the so-concatenated histogram is less-fluctuant and more reliable for eye representation. The eLBPH is less variational, and more reliable to extract the ocular information.

1. The Gabor features extract the eye features in different orientations and different scales, but they give a coarse account of the global shape information (global texture representation), and not local texture information. Moreover, Gabor feature vector is slightly long even after down-sampling.
2. The LTP realizes a good results in current datasets. This is due to that LTP features give a detailed account of the appearance of ocular regions while resisting to the lighting variations and overcome the image noise. However, unlike LBP an effective pre-processing chain is used to greatly reduce the influence of different variations (illumination). We observe also that the LTP histogram dimensionality is very large, and through our experiments LTP is far to be efficient without pre-processing.

From conducted experiences, either ePLBPH* or eLBPH^{u2} achieved a good performance for eye localization. However, eLBPH^{u2} proves to be insufficient for describing the eye textures in different resolutions, since textures in low resolution are coarsely represented by the LBPs because, they contain fewer texture details of the eye pattern while textures in high resolution are more detailed. The eye textures are comparatively easy to be discriminated at specific resolution than others. So, combining the texture information of various resolutions, contributes for

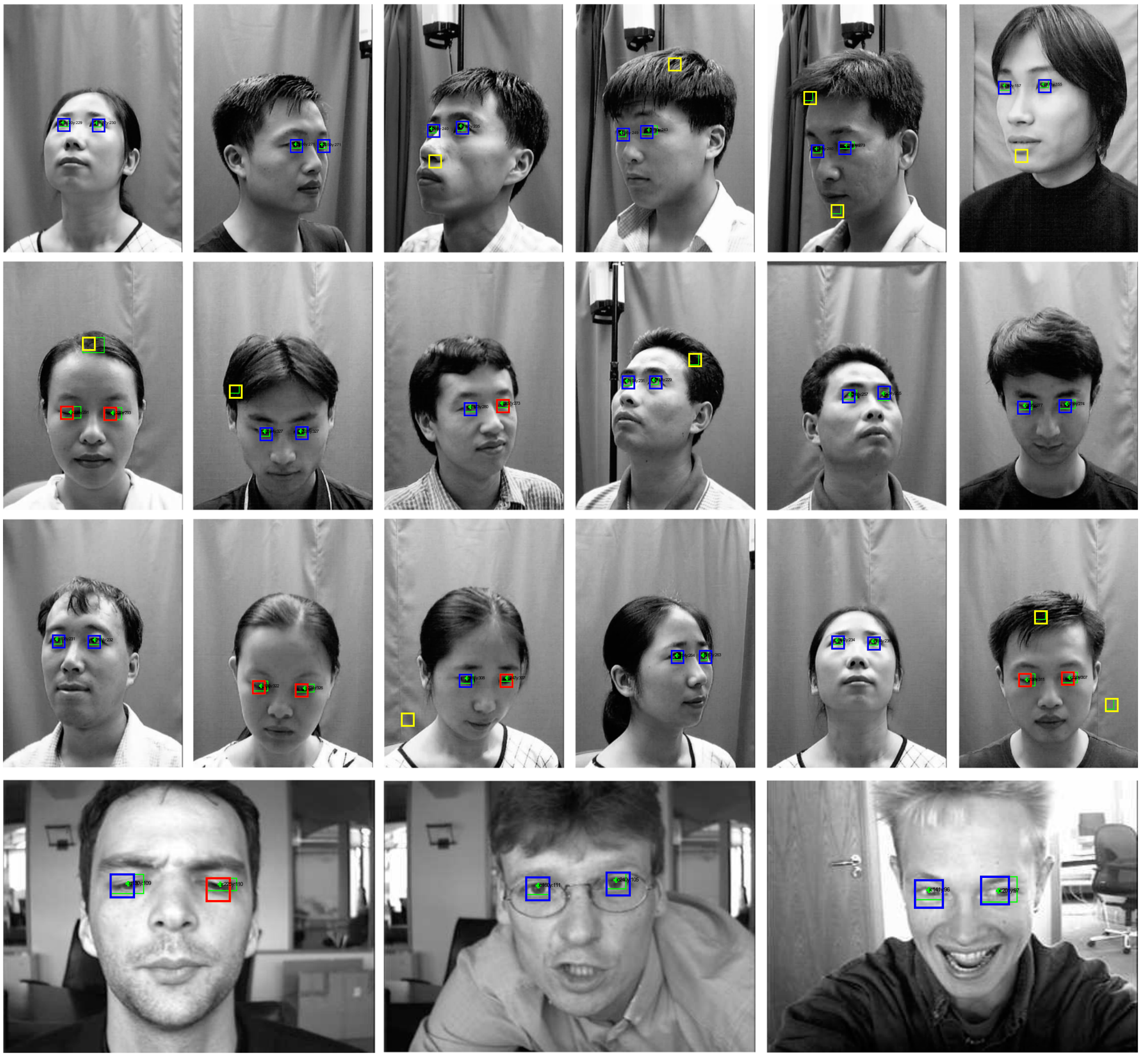


Figure 7 Some eye localization in challenging cases with hard variations in pose, facial expression.

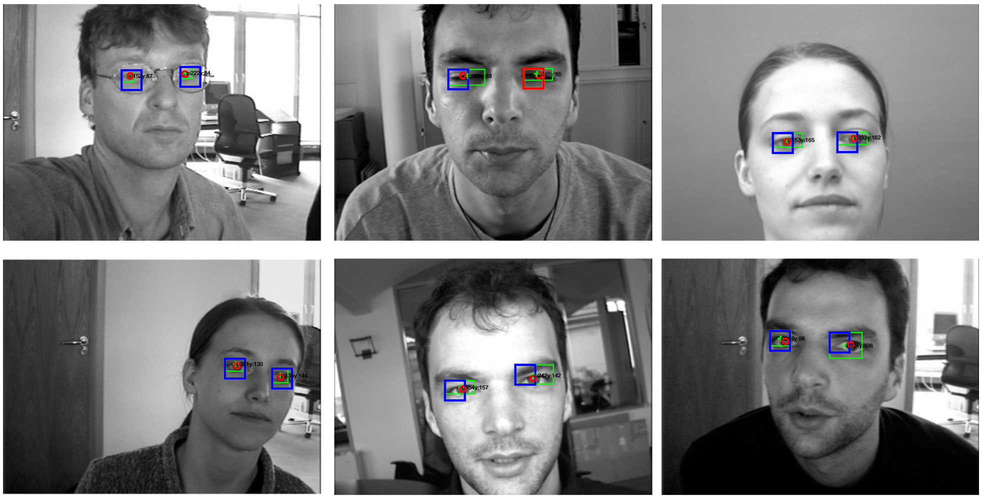


Figure 8 Example of some successful eye localization of our method (BioID database): variations in pose, facial expression, subject is wearing glasses.

their discrimination and enhances the effectiveness of the used features descriptor. The proposed ePLBPH* fulfills this requirement and shows excellent results for detecting and locating eyes under hard nature settings. Figures 6a and b give the ROC curves of considered features on BioID and CAS-PEAL-R1 datasets for eye localization. In most cases it can be seen that the ePLBPH* features realized one of the best performance in terms of AUC values with SVM and MLP models, with the previous discussed results. We can observe that the ePLBPH* is beneficial for detecting and locating the eyes.

The used datasets contain facial images with variation in expression, head pose, illumination and views (resolution changes). The visual inspection is used as an evaluation measure for demonstrating the robustness of our methods against these variations.

The results listed in Tables 2 and 3 demonstrate that our method gives excellent detection accuracy on the BioID and the CAS-PEAL datasets. More stringent evaluation tests are shown by examples of successful location of eyes on current datasets Figs. 7, 8, and 9. The obtained eye location is the green cross and the ground-truth of the real eye locations is the red circle, within the pupil radius (provided from the BioID and manually annotated in CAS-PEAL datasets). The different bounding-box colors correspond to various matching objects, detection results cover the right location of the eye and false alarms are the missed positive; the blue rectangle is an eye open, the red rectangle means an eye closed, these results from the $\mathbb{S}_{\chi^2}$ measure, which is used to verify the presence of eyes and not their state. The yellow rectangles are the classifier false positive, rejected by the $\mathbb{S}_{\chi^2}$ measure (considered an insignificant information by the algorithm) (Figs. 10 and 11).

5.3.5 Comparison of EyeLSD with Recent Works

By the lack of common benchmark datasets and accepted assessment protocol, making a fair comparison between

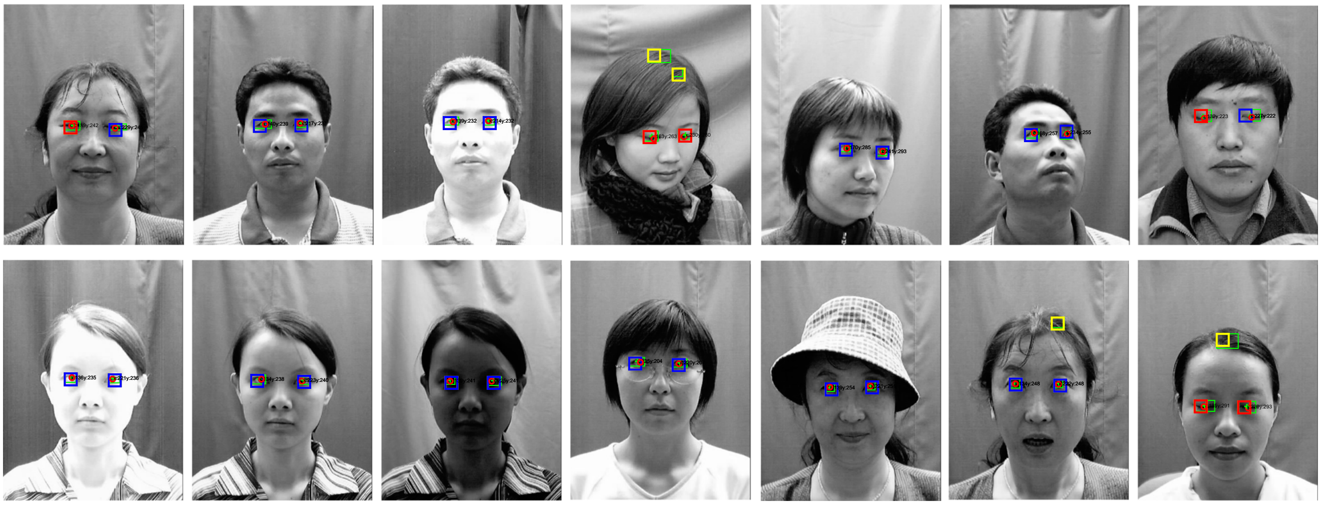


Figure 9 Eye localization in CAS-PEAL database: the *green cross* corresponds to the output of our system and the *red* circularly form is the ground truth of the real eye coordinates, this needs to be marked manually in CAS-PEAL-R1 database.

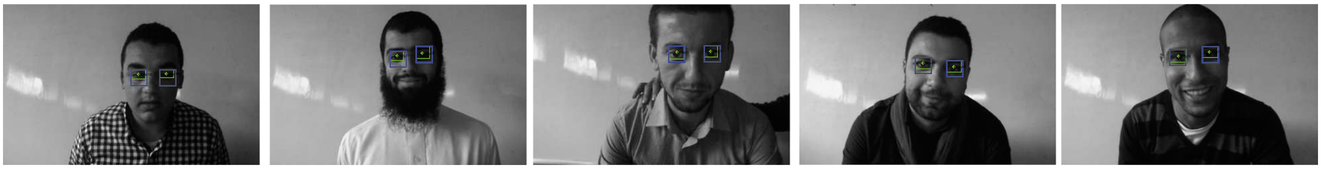


Figure 10 Snapshots illustrate some successful detection on pictures captured within the laboratory.

EyeLSD approach and existing ones in the literature, is not a simple issue.

In spite of this, Table 4 shows most recent eye localization methods we are aware, beside of the proposed one. Withal, corresponding experimental settings are presented such as, image dataset, beside of its characteristics (number of test images with their corresponding size) and the final performance realized by each algorithm.

The overall performance of the proposed eye location estimation scheme is similar, and in some cases better than algorithms of comparison.

The common points between our eye detector and the listed methods (Table 4) is that the presented approaches depend on the appearance and image patch based methods, either by using supervised (SVM and MLP) or unsupervised learning methods (Boltzmann machine model and Independent Component Analysis (ICA)). As well as, these methods are using the same benchmark image gallery, that challenge the real-world conditions.

In [15], authors investigate the problem of eye localization for subjects wearing glasses under different constraints. This method is based on Variance Filter (VF) that measures the gray intensity change of the eyes and ICA applies to recognize the right eye location. The eye center is searched within detected ocular regions, by calculating the entropy of intensity change. Hassaballah et al. [15] method realized a detection rate of 97.1 % with 600 test images of BioID dataset. Nonetheless, the detection fails if the light reflection on glasses is too strong and occlusions occult the facial region.

In contrast, the proposed framework surpasses aforementioned methods, realized excellent results and capable of discriminating the ocular region under real challenging conditions.

In [44], they are proposing an eye detection framework able to estimate the eye region location with precision. The input image is pre-processed to highlight the eye structure, that is used after the localization step. So, the eye pair is extracted by using binary template matching and SVM classifier. The next step consists to accurately detect the eyes by using VF. This algorithm is trained with 800 images collected from BioID database, and yielded a detection rate of 95.6 % . However, the detector fails under hard illumination conditions, strong light reflection and closed eyes. This occurs mostly because the template matching step fails to find the right eye pair location. In [47], authors proposed a deep-learning algorithm for detecting the eyes under uncontrolled conditions. The robustness of the detection scheme is tested on different datasets and in different conditions (facial expression, low-resolution, pose, and illumination). The potential of [47] approach to handle the resolution variations, is assessed on BioID dataset. The eye images are evaluated with their original size and with images down-sampled to 50 %. So, two resolutions of eye image patches are generated and used to show ability of the deep features trained model, to recognize the right eye location despite low-resolution images.

Boltzmann-deep features learner is tested on 956 images of BioID dataset, and achieves a competitive results to those of the LBP features with Viola-Jones eye detector. The main advantage of Viola-Jones approach is its computational effectiveness. However, the performance of that method depends on the amount and diversity of the training data. So, it may not give a right eye location. Meanwhile,

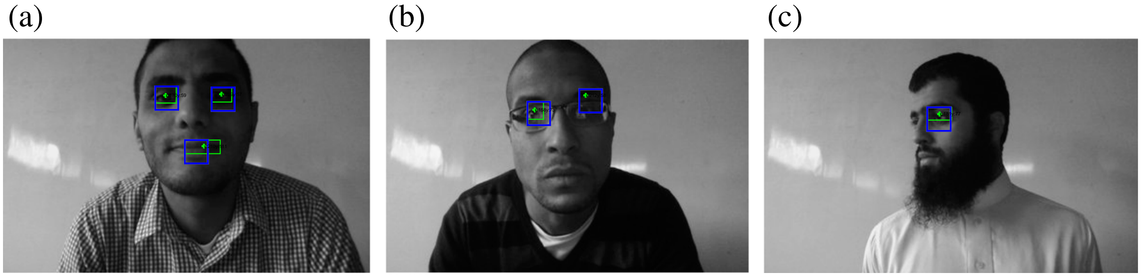


Figure 11 Snapshots from captured pictures illustrating some typical failures of our method: individuals with different head rotation angles, facial expressions and with accessories (myopia glasses, beard, expressions).

Table 4 Comparison of EyeLSD with existing methods.

Method	Database	challenge	patch size (pixels)	# Test images	Acc. (%)
EyeLSD with ePLBPH*(Proposed)+ MLP [4]	BioID	Variation in distance (multi-resolution) Illumination condition, occlusion, gender, aging, expression, hard rotation	**24 × 24**	969	**98.12**
EyeLSD with ePLBPH*+ SVM(RBF)	BioID	Variation in distance (multi-resolution) Illumination condition, occlusion, gender, aging, expression, hard rotation	**36 × 36**	969	**98.86**
VF + ICA [15]	BioID	Variations in views, lighting conditions, occlusions aging, expressions	60 × 30	600	97.1
Learning the Boltzmann [47] Machine model	BioID	Expression, illumination, pose, low resolution	**36 × 36**	956	**98.12**
LBP [47], Viola-Jones [43]				956	**98.64**
Learning the Boltzmann [47] Machine model	BioID	Expression, illumination, pose, low resolution	18 × 18	956	98.12
LBP [47], Viola-Jones [43]				956	98.01
VF + SVM [44]	BioID	Dynamic background, moderate rotation, glasses wearing and face occlusions	25 × 8		95.6
SIFT features + SRC [33]	BioID	Slight variation in pose illumination changes various background face sizes image rotations expressions	60 × 60	1000	91.5

building a classifier that learns the variability of eyes might meet with problems, and even if a large set of training image is used.

By using a descriptor window enlarged to 36 × 36 pixels, we are able to surpass performance realized by LBP [47] and Viola-Jones [43] methods. Intuitively, a larger eye patch of 36 × 36 pixels has more discriminative information, and thus, reduces the false positive rate but it will be at the cost of losing generalization ability for locating eyes, and the extracted features will be less likely to be good representative of eyes [37].

Furthermore, our method shows robustness against the rotation of face area and extreme pose, while that most errors occurred in the Viola Jones eye detector, are related to the facial textures rotations and head pose variations.

From Table 4, the results of the proposed approach are comparable with those obtained by Boltzmann machine model and LBP features with Viola-Jones methods for eye localization.

In [33] authors proposed a learning method for eye localization in arbitrary rotation settings. They examine the feasibility of localizing eyes without prior face detecting. A pyramid-like eye locating strategy is used for coding local-features, ensured with SIFT descriptor and the sparse representation classifier (SRC), classifies the image patches through input image. Then, a searching map called (HeatMap) is generated from the adjusted classifier's outputs, and the potential positions of eyes are highlighted. (The HeatMap highlights eye centers by superposing the adjusted classifier's output values through Pyramid-like method). To locate the eye center, while reducing noise effect and influence of complex backgrounds, the skin color algorithm is applied in HSV color space, that improves skin detecting and isolates the facial region from the background. The false positives detected around the real center of eye positions are rejected by using similarity function and the center of eyes are retained if a maximal similarity score is reached. Their method are assessed on several datasets. For a fair comparison with our approach, we consider only tests performed on BioID database, which achieve an accuracy of 91.5 % on 1000 image patches of size 60 × 60. We observe that the accuracy of the proposed method is better than the method of [33], tested on 969 images of same dataset. Among all the the listed methods, our ePLBPH* achieves the highest classification score on BioID dataset. This enhancement is attributed to the combination of multiple LBP-mapping schemes and the application of spatial enhanced pyramid-like image decomposition strategy. The ePLBPH* tolerates several changes, illumination, image blur, perspective and rotation. So, the proposed eye detector is robust against the multiple variations, that the presented works shown their distinctiveness performance limitations, Figs. 7, 8, and 9.

5.4 Eye State Detection Experiments

In this section, we first introduce a real-world dataset for algorithm verification. The performance of different feature descriptors on precedent datasets described in Section 5.2, under the proposed framework (eye state detection) presented in Section 3. Then, a detailed investigation about the performance realized by our method is conducted. Finally, to verify the effectiveness of the proposed multi-scale extension of TPLBP, we describe and compare most recent methods with the newly proposed one. In this part, we consider two types of feature descriptors (patch-based and pixel-based LBP features), which can capture local and global texture information even under challenging conditions. In particular, we use the proposed extension of TPLBP histograms (Multi-TPLBP) described in Section 4.2.1 and multi-scale extension of two pixel-based LBP feature sets. The obtained results are given in Table 5.

In this experiment uniform pixel-based LBPs are extended to the multi-resolution representation and compared against performance realized by the patch-based LBP descriptors. In the proposed TPLBP extended to the hierarchical multi-scale sampling (Multi-TPLBP), the radius of the TPLBP rings is enlarged twice as well as joint to the LPF. During the multi-resolution image generation ($R_x = R_y = 1$), that means no down-sampling is made. Several constraints should be respected during descriptor construction as described in Section 4.2.1. The 3-scale patch- and pixel-based descriptors are built with eight sampling points, and a radius values of $\mathbf{R} \in \{1.0, 2.4, 5.4\}$.

Table 5 gives a comprehensive classification performance comparison of precedent feature sets in terms of detection accuracy (Acc), TP(resp. TN), FP(resp. FN) and AUC. In this experience, TP (resp. TN) is the percentage of instances of eye open class (resp. eye closed class) well classified, while FP(resp. FN) is the percentage of instances of eye open class (resp. eye closed class) miss-classified. Several observations are made from this table (Table 6).

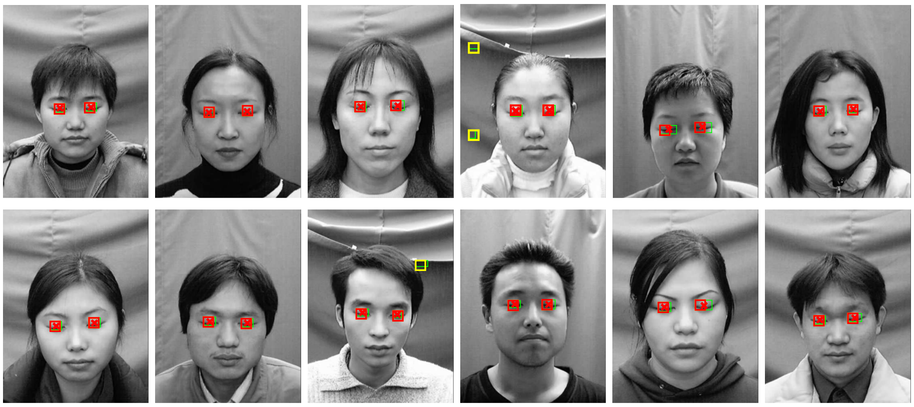
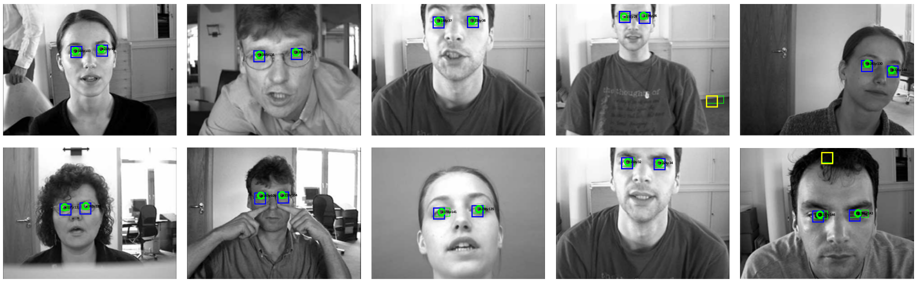
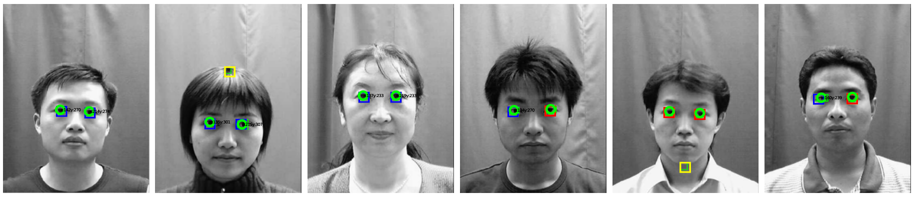
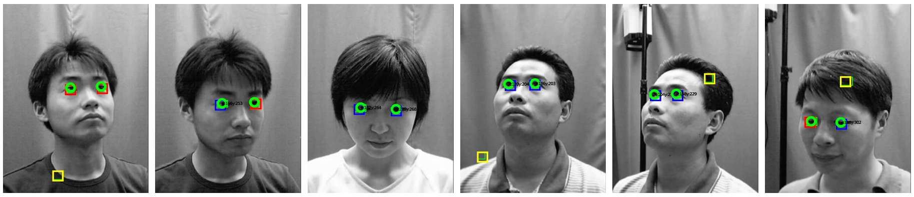
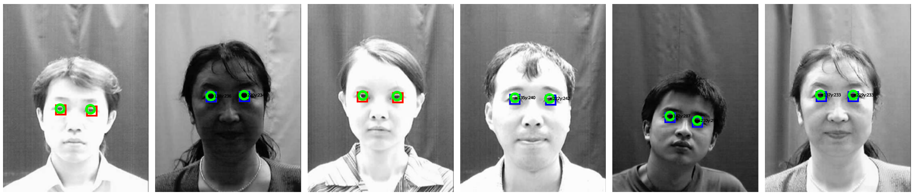
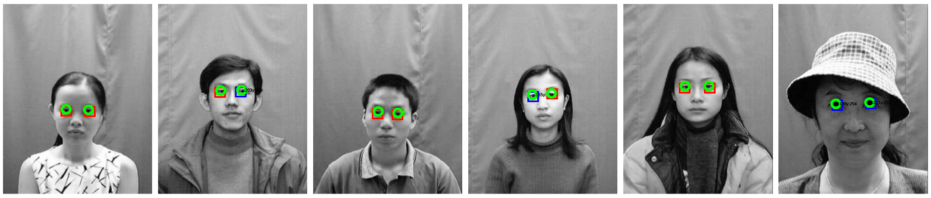
1. we can see that the Multi-LBP^{u2} realized an enhancement over the performance of Multi-LBP^{riu2}, which is about 10.07 %, 10.8 % and 9.82 % with SMV(linear), SVM(RBF) and MLP, respectively. At the first sight, a reasonable improvement of Multi-LBP^{u2} is realized, compared to the Multi-scale LBP^{riu2}.

   One difference can, however, arise between these two pixel-based feature sets, that is attributed to the feature vector length. Also, despite being invariant to rotation (the operator is tolerant to the texture rotation), Multi-LBP^{riu2} captures the uniform texture information even if rotation happens, and supports the major part of the texture information. However, the descriptor is too short and maybe the eye texture can be not reliably represented.

Table 5 Eye state: statistical results on ZJU database.

Method	TP (%)	FP (%)	TN (%)	FN (%)	Prec	Rec	F_1 *Score*	Acc (%)	AUC (%)
SVM(Linear)									
TPLBP	96.50	51.70	48.29	3.49	0.85	0.96	0.90	84.45	85.92
Multi-TPLBP	96.09	8.29	91.70	3.90	0.97	0.96	0.97	95.00	97.85
Multi-LBP^{u2}	96.09	18.29	81.70	3.90	0.94	0.96	0.95	92.50	97.45
Multi-LBP^{riu2}	96.74	60.48	39.51	3.25	0.83	0.97	0.89	82.43	86.94
SVM(RBF)									
TPLBP	96.01	49.51	50.48	3.98	0.85	0.96	0.90	84.63	86.82
Multi-TPLBP	96.34	8.29	91.70	3.65	0.97	0.96	0.97	95.18	97.83
Multi-LBP^{u2}	95.04	10.97	89.02	4.95	0.96	0.95	0.96	93.54	97.73
Multi-LBP^{riu2}	95.44	55.36	44.63	4.55	0.84	0.95	0.89	82.74	87.00
MLP									
TPLBP	96.01	47.80	52.19	3.98	0.86	0.96	0.91	85.06	87.63
Multi-TPLBP	96.17	8.53	91.46	3.82	0.97	0.96	0.97	95.00	98.12
Multi-LBP^{u2}	94.30	11.95	88.04	5.69	0.96	0.94	0.95	92.74	97.40
Multi-LBP^{riu2}	94.47	51.70	48.29	5.52	0.85	0.94	0.89	82.92	86.59

Table 6 Examples of some typical success our eye open/closed approach, tested on different conditions (Pose, lighting, resolution, facial expression, occlusion).

Challenge	Detection results
Eye closed CAS-PEAL	
Eye open BioID	
Eye open CAS-PEAL	
Pose	
Illumination	
Resolution accessory	

The 3-scale LBP^{riu2} (Multi-LBP^{riu2}) generates a histogram of 10 bins × 3 = 30 bins, which is less than six times the length of the Multi-LBP^{u2} histogram of 177 bins. In addition, uniform LBP can represent the most local structures of the eye image which are represented by uniform codes, while the noise patterns

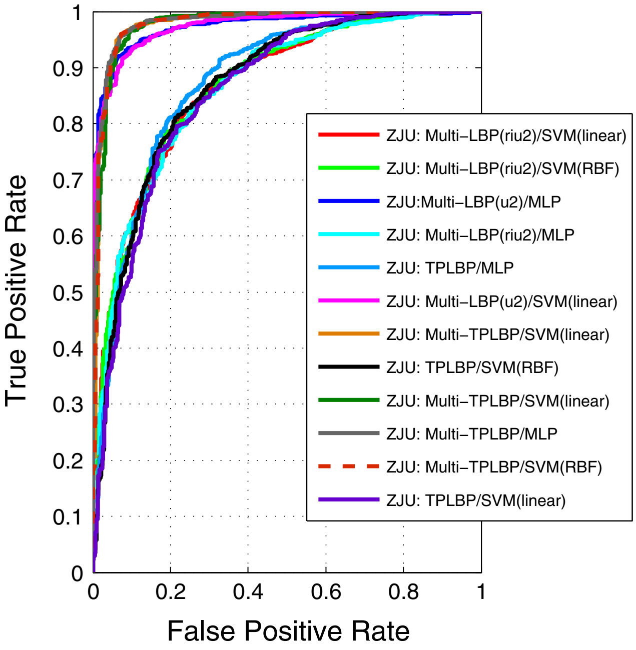


Figure 12 ROC curves of various features using the SVM and MLP classifiers on ZJU Eyeblink dataset.

most likely fall into the non-uniform codes. The extension of this descriptor in the multi-resolution domain allows the descriptor to capture additional information than the basic representation. In that way Multi-LBP^{u2} extracts the local information about the eye state, the global shape of the eyes, and the deformation of their textures. However, some image patterns such as lines are not captured in uniform codes [32]. These line patterns may appear less frequently than uniform codes, but they represent a set of important local primitives for pattern recognition.

2. The Multi-TPLBP generates a performances gained over those of TPLBP of 10.55 %, 10.55 % and 9.94 % with SMV(linear), SVM(RBF) and MLP, respectively. The poor imaging conditions, such as low-resolution, blur, Gaussian noise, and uneven light are leading to ambiguous appearance of the eyes and specifically difficult to differentiate an eye state from another, as shown in Fig. 5. The obtained performance proves that TPLBP cannot handle well all these variations. So, TPLBP realized a low accuracy of 85.06 %. TPLBP describes well the eyes in open state with 96.01 % of open eyes correctly classified, but only 52.19 % of closed eyes are classified well. One possible explanation, when the eyes are screwed up, it is difficult to TPLBP to describe the appearance of closed eyes with a coarse account of the global shape information.
3. The Multi-LBP^{u2} outperforms the TPLBP descriptor with classification score 93.54 %. LBP^{u2} shows an improved performance for the eye state description and the 3-resolution fusion approach compared to the TPLBP. One reasoning can be, the pixel-based computation of LBP captures texture variations minimally when compared with the TPLBP descriptor.
4. Over the precedent comparison, we can observe that the proposed Multi-TPLBP improves the performance upon its original version. In ZJU dataset, the Multi-TPLBP clearly outperforms precedent descriptors, with a best accuracy of 95.18 % with SVM(RBF) and an AUC value of 97.83 %. Figure 12 gives the ROC curves of those features. It can be seen that the Multi-TPLBP feature realized best performance in terms of AUC values with SVM and MLP models, followed by Multi-LBP^{u2}, which further verifies that our patch-based LBP extensions is beneficial to eye state detection task.

Figure 13, illustrates different failure of the proposed approach. It can be noticed that Multi-TPLBP fails on extracting the eye features under hard angles of head rotation, in case of the eye is not detected at all, or the perspective changes even when the eye is well localized. One possible explanation, the Euclidean distance score used to build TPLBP [45], fails under the precedent circumstances (variations in depth scales), and hence, detecting the correct eye state may also fail. Figures 6 and 13 show the eye state estimation results and false alarms generated by the algorithm. It can be noticed that the change in facial expression results in degradation of the recognition performance of patch-based LBP descriptor. In the real-world applications

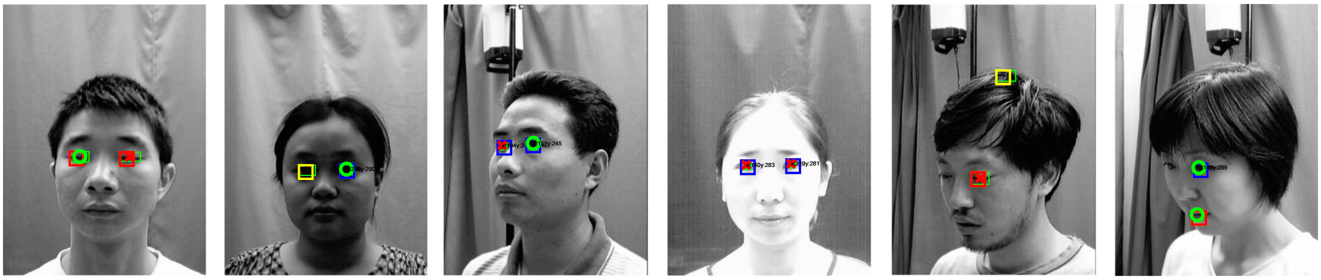


Figure 13 Snapshots illustrating some typical failures of our eye state detection method: it includes individuals with different head rotation angles, illumination variation.

such as the detection of the driver drowsiness and the weak attention paid to the road, facial images are not always captured in frontal view. This introduces a pose variation with the ocular region variations and occlusions depending of the head-pose.

We can observe, that by lowering eyebrows while exhibiting expression changes the ocular region. In Fig. 6, it is noticed that there is no performance degradation of the eye state estimation for smiling scenario tests. In Fig. 6 and 13 it can be seen that large variations in pose significantly affect performance of the eye state estimation. When the face is turned, out-of-plane rotation, parts of the ocular region towards the direction of the rotation becomes hidden while to other side of the face undergoes skew and sometimes a degradation of the eye texture due to the light variations.

Figure 6 clearly shows that the Multi-TPLBP descriptor handles well perspective variations and correctly recognizes the eye states, without employing any face frontalization algorithm [16]. From Fig. 6 we can notice that the algorithm detects well the location of the eyes, but the state of the eye is not correctly estimated under different lighting conditions. This is due to the image regions captured in different modality, with a very different pixel intensity values. The description of these regions implies mismatches of the similarities between neighboring patches of pixels (self-similarity calculation to generate the TPLBP code), and hence our algorithm detects an eye closed instead of open.

5.4.1 Comparison of the State Model with Other Methods

Table 7 compares the proposed EyeLSD and recent works we are aware tested on ZJU dataset, with corresponding experimental settings (number of eye open (+) and eye closed (-), the number and the size of the eye images used during tests, and challenges faced), and the performance realized by each approach is listed beside ours.

Pan et al. present in [32] an appearance-based eye blink detection application, the ZJU Eye blink dataset is used to carry on tests. The performance realized by their framework is 93.3 %. The proposed Multi-TPLBP outperforms this approach with an accuracy of 95.18 %.

Multi-TPLBP captures more texture information complementary to those of the LBP-based pixels. This is reflected through the performance gain. In addition, it is invariant to scales and that forms a great need to build a robust eye state model that resists well to the real-world constraints. In such scenarios the eyes may undergo various poses and scales and not really obvious for some local shape descriptors, to neutralize the effects involved by those variations.

Fengyi Song et al. present in [38] two methods to extract the eye features under real-world conditions. Their eye state approach is tested with a first method called Histograms of principal Oriented Gradients (HPOG) and a second one called Multi-scale Histograms of principal Oriented Gradients (MultiHPOG). Their approaches were assessed on ZJU eye blink database, under different variations of facial expression, lighting, individual identity, and image noise. Their system architecture includes geometric alignment, which is considered as a key point of the algorithm. In Table 7 the performance realized by the Histograms of HPOG without alignment are shown, including geometric normalization of the eye image patches. In case of no alignment used, HPOG achieved a recognition accuracy of 94.04 %. However, when the alignment is used, recognition performance is enhanced with 1.87 %. The performance enhancement realized by Multi-TPLBP, is partly owed to the additional local information captured from the enlarged

Table 7 Comparison of the eye state model with existing methods.

Method	Data	challenge	patch size (pixel)	# Test (+) open, (-)closed	Acc (%)
Multi-TPLBP(Proposed)+ MLP	ZJU	Varying in distance, Illumination condition, occlusion, gender, aging, expression, hard rotations	24 × 24	1230(+), 410(−)	95.18
LBP + SVM [32]	ZJU	varying pose, lighting, accessory.	0.74 × 0.37 (ratio to eyes distance)	Rear clips (clip #3, clip#4)	90.37 84.37
HPOG (without alignment) + SVM [38]	ZJU	Variations in pose, lighting, accessory	24 × 24	1230(+), 410(−)	94.04
HPOG (alignment) +SVM [38]					95.91
MultiHPOG +SVM [38]					95.60
MultiHPOG/LTP/Gabor + SVM [38]					96.83

feature extraction area of the eye pattern and the texture perimeter that the descriptor can reach with its enlarged TPLBP radius. In addition, applying Gaussian low-pass filters to construct the multi-resolution descriptor, attenuates the image noise effect and increases the contextual information amount captured. EyeLSD outperforms the HPOG (without alignment) approach and realized a slightly worse accuracy in case of aligned eye image patches.

Fengyi Song et al. [38] investigate the enhancement of feature fusion to describe images under uncontrolled conditions. They combine Haar-like feature approach, Multi-HPOG, LTP, and Gabor wavelets to extract salient eye feature map. These methods can complement each others in terms of information capturing and imaging conditions resistance. At the first sight, Gabor feature sets are an optimal tool used for the purposed of local feature extraction. Salient visual proprieties of the eyes including spatial localization, orientation selectivity and spatial frequency selectivity are quit described. However, Gabor wavelets is far too computationally expensive. The multi-scale HPOG is instead able to represent the eye image patch in varying scales. Thus, captures the eye appearance at different scales and further information that is normally missed by local descriptors.

The fusion of feature sets can enhance the overall system accuracy, that realized 96.83 % on ZJU database. These results are slightly better than those realized by Multi-TPLBP, tested on the same image gallery. Nonetheless, fusion of feature sets involves a high calculation complexity with a small improvement realized compared to use of Multi-HPOG only.

To further verify the effectiveness of the proposed method and besides performance evaluation on ZJU dataset. The most recent related states-of-the-art eye state detection methods, are presented but not directly compared with our approach. Either the approaches or the used datasets for validation are different from ours. So, a fair and direct comparison is not possible. The generalization capability of the Multi-TPLBP is demonstrated on tests conducted on BioID and CAS-PEAL-R1 datasets. Even since we did not train the eye state models on those image sets, but only running our algorithm on them, obtained results are illustrated on Fig. 6.

It should be noticed that, by combining into the ePLBPH structure the strength of pixel-based LBP approaches (for local description) with that of histogram concatenation (global information encoding), and multi-level pyramidal architecture (multi-scale information captured under various depth), the performance of the eye detection algorithm is significantly enhanced. Furthermore, Multi-TPLBP may fail to detect eyes displayed at different depth scales and even if a pyramidal structure is applied to the TPLBP descriptor, the increase in complexity may hinder further the operator performance. As shown in [25] the TPLBP extended to pyramidal architecture is sensitive to facial expression and head pose, which may result in lower performance for the eye detection step. However, compared to the LBP variants, the proposed Multi-TPLBP framework provides good results in the eye state detection problem, where the challenges are less restrictive.

5.5 Runtime Performance Evaluation

The overall run-time of our MATLAB implementation is performed on an Intel Core i7-4790 Processor with 3.6 GHz and 8.0 GB Ram. We run the EyeLSD on an image of 360 × 480 pixels. The average computation costs of the principal EyeLSD stages are listed in Table 8, that reports the average elapsed time at each processing step. These steps must run sequentially in each key-point of the pre-processed image. From Table 8, we can observe that the pre-processing step represents an embarrassingly parallel workload, since each key-point and neighborhood pixels are scanned, which span the entire local minimum regions of the input image. The table reveals that the feature extraction and classification steps take about 30 % of the total time, while 70 % of the time is due to the pre-processing step. In practice, the pre-processing step can be replaced with a facial landmark localization algorithm [36], that precisely locates different facial traits and highlights them as a landmarks (e.g., the nose tip, mouth corners, eye centers) in the input image. So, instead of searching for an eye in the entire image over key-points. we can only scan the facial landmarks to locate the eye positions and recognize whether open or closed. This process can save the time spent on the steps of pre-processing, and exploits more the potential of the proposed feature extractors. Moreover, by using other low-level programming language such as C++ and beside of code optimization strategies, the algorithm computation time can be further enhanced.

Table 8 Computation time of each step of EyeLSD approach.

Pre-processing (ms)	Eye localization (ePLBPH*)(ms)	Eye localization (Prediction)(ms)	$\mathbb{S}_{\chi^2}$(ms)	Eye state (Multi-TPLBP)(ms)	Eye state (Prediction) (ms)	Total (ms)
112.89	14.9	9.20	0.41	20.7	3.0	161.1

6 Conclusion

Eye localization and state estimation is required in a wide range of vision-based applications. Including driver drowsiness detection and disabled person assistance systems, but those problems are far from being solved, especially in uncontrolled environments. As a contribution to this problem, in this paper we have proposed the EyeLSD for eye localization and its state estimation (open or closed). The EyeLSD performance was compared against existing state-of-the-art approaches.

The EyeLSD (with ePLBPH*) has achieved an accuracy of 98.12 % for eye localization on BioID dataset. Comparing this accuracy against the reported by other approaches, it is possible to conclude that our approach is ranked on the top two with higher accuracy. This accuracy is reached due to ePLBPH*, which has higher tolerance to illumination, image blur and perspective changes. This makes EyeLSD robust against multiple variations where other approaches fail, Figs. 7, 8, and 9.

In contrast to eye localization problem, the eye state detection problem has very few works evaluated under a common public dataset. However, our evaluation of EyeLSD accuracy to detect eye state shows promising results. The proposed Multi-TPLBP has so far the best recognition accuracy for eye state estimation on ZJU database. Indeed Pan et al. present in [32], an appearance-based eye blink detection application, which was evaluated under the ZJU dataset. Comparing the accuracy under this dataset, their approach got 93.3 % and EyeLSD with Multi-TPLBP got 95.18 % which outperforms their approach. This high accuracy is obtained, because Multi-TPLBP captures more texture information complementary to those of the LBP-based pixels and it is invariant to scales. In summary, the proposed Multi-TPLBP can solve those drawbacks by enlarging the perimeter of information capturing, while reducing noise and redundant information with filtering process.

As future work we intend to implement EyeLSD in a low cost computational platform for real time safety applications and explore its implementation in a Graphical Processing Unity (GPU) or FPGA.

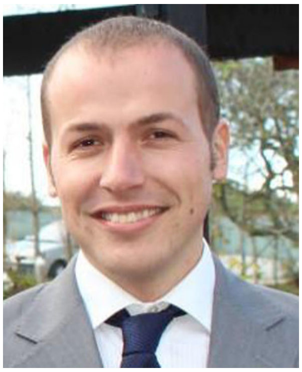
Acknowledgments Project NORTE-01-0145-FEDER-000020” is financed by the North Portugal Regional Operational Programme (NORTE 2020), under the PORTUGAL 2020 Partnership Agreement, and through the European Regional Development Fund (ERDF).

References

1. (1998). PERCLOS: a valid psychophysiological measure of alertness as assessed by psychomotor vigilance. In *Indianapolis, Federal Highway Administration, Office of Motor Carriers*. Tech. Rep. MCRT-98-006.
2. Ahonen, T., Hadid, A., & Pietikainen, M. (2006). Face description with local binary patterns: application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(12), 2037–2041.
3. Benrachou, D.E., Boulebtateche, B., & Bensaoula, S. (2013). Gabor/pca/svm-based face detection for drivers monitoring. *Journal of Automation and Control Engineering*, *1*, 115–118.
4. Benrachou, D.E., dos Santos, F.N., Boulebtateche, B., & Bensaoula, S. (2015). Automatic eye localization multi-block lbp vs. pyramidal lbp three-levels image decomposition for eye visual appearance description. In *Pattern recognition and image analysis* (pp. 718–726). Springer.
5. Benrachou, D.E., dos Santos, F.N., Boulebtateche, B., & Bensaoula, S. (2015). Online vision-based eye detection: Lbp/svm vs lbp/lstm-rnm. In *CONTROLO'2014–proceedings of the 11th Portuguese conference on automatic control* (pp. 659–668). Springer.
6. Chang, C.C., & Lin, C.J. (2011). Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *2*(3), 27.
7. Cootes, T.F., Taylor, C.J., Cooper, D.H., & Graham, J. (1995). Active shape models-their training and application. *Computer Vision and Image Understanding*, *61*(1), 38–59.
8. Feng, G.C., & Yuen, P.C. (1998). Variance projection function and its application to eye detection for human face recognition. *Pattern Recognition Letters*, *19*(9), 899–906.
9. Flores, M.J., Armingol, J.M., & de la Escalera, A. (2010). Real-time warning system for driver drowsiness detection using visual information. *Journal of Intelligent & Robotic Systems*, *59*(2), 103–125.
10. Gao, W., Cao, B., Shan, S., Chen, X., Zhou, D., Zhang, X., & Zhao, D. (2008). The cas-peal large-scale chinese face database and baseline evaluations. *IEEE Transactions on Systems, Man and Cybernetics Part a Systems and Humans*, *38*(1), 149.
11. Ge, S., Yang, R., Wen, H., Chen, S., & Sun, L. (2014). Eye localization based on correlation filter bank. In *2014 IEEE international conference on multimedia and expo (ICME)* (pp. 15). IEEE.
12. Gonzalez, R.C., & Woods, R.E. (2002). *Digital image processing*, 2nd edn. (pp. 523–532). Englewood Cliffs, NJ: Prentice-Hall.
13. González-Ortega, D., Díaz-Pernas, F., Antón-Rodríguez, M., Martínez-Zarzuela, M., & Díez-Higuera, J. (2013). Real-time vision-based eye state detection for driver alertness monitoring. *Pattern Analysis and Applications*, *16*(3), 285–306.
14. Harwood, D., Ojala, T., Pietikäinen, M., Kelman, S., & Davis, L. (1995). Texture classification by center-symmetric auto-correlation, using kullback discrimination of distributions. *Pattern Recognition Letters*, *16*(1), 1–10.
15. Hassaballah, M., Kanazawa, T., & Ido, S. (2010). Efficient eye detection method based on grey intensity variance and independent components analysis. *Computer Vision, IET*, *4*(4), 261–271.
16. Hassner, T., Harel, S., Paz, E., & Enbar, R. (2015). Effective face frontalization in unconstrained images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4295–4304).
17. Hollingsworth, K., Clark, S., Thompson, J., Flynn, P.J., & Bowyer, K.W. (2013). *Eyebrow segmentation using active shape models, SPIE Defense, Security, and Sensing* (pp. 871,208–871,208). International Society for Optics and Photonics.
18. Kalbkhani, H., Shayesteh, M.G., & Mohsen Mousavi, S. (2013). Efficient algorithms for detection of face, eye and eye state. *Computer Vision, IET*, *7*(3), 184–200.
19. Lazebnik, S., Schmid, C., & Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (Vol. 2, pp. 2169–2178). IEEE.

20. Lehmann, A., Leibe, B., & Van Gool, L. (2011). Fast prism: Branch and bound hough transform for object class detection. *International Journal of Computer Vision*, *94*(2), 175–197.
21. Leibe, B., Leonardis, A., & Schiele, B. (2008). Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision*, *77*(1-3), 259–289.
22. Li, W., Wang, Y., & Wang, Y. (2011). Eye location via a novel integral projection function and radial symmetry transform. *International Journal of Digital Content Technology and its Applications*, *5*(8), 70–80.
23. Mäenpää, T. (2003). *The Local binary pattern approach to texture analysis: extensions and applications*. Oulun Yliopisto.
24. Mäenpää, T., & Pietikäinen, M. (2003). Multi-scale binary patterns for texture analysis. In *Image analysis* (pp. 885–892). Springer.
25. Mahalingam, G., & Ricanek, J.r. K. (2013). Lbp-based periocular recognition on challenging face EURASIP. *Journal on Image and Video Processing*, *2013*(1), 1–13.
26. Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, *29*(1), 51–59.
27. Ojala, T., Pietikäinen, M., & Mäenpää, T. (2000). Gray scale and rotation invariant texture classification with local binary patterns. In *Computer vision-ECCV 2000* (pp. 404–420). Springer.
28. Ojala, T., Pietikainen, M., & Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(7), 971–987.
29. Oyini Mbouna, R., Kong, S.G., & Chun, M.G. (2013). Visual analysis of eye state and head pose for driver alertness monitoring. *IEEE Transactions on Intelligent Transportation Systems*, *14*(3), 1462–1469.
30. Pang, Y., Yuan, Y., & Li, X. (2008). Gabor-based region covariance matrices for face recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, *18*(7), 989–993.
31. Qian, X., Hua, X.S., Chen, P., & Ke, L. (2011). Plbp: An effective local binary patterns texture descriptor with pyramid representation. *Pattern Recognition*, *44*(10), 2502–2515.
32. Pan, G., Sun, L., Wu, Z., & Lao, S. (2007). Eyeblink-based anti-spoofing in face recognition from a generic webcamera. In *IEEE 11th international conference on computer vision, 2007. ICCV 2007* (pp. 1–8). IEEE.
33. Ren, Y., Wang, S., Hou, B., & Ma, J. (2014). A novel eye localization method with rotation invariance. *IEEE Transactions on Image Processing*, *23*(1), 226–239.
34. Shi, Y., Yan, Z., Ge, H., & Mei, L. (2014). Visual objects location based on hand eye coordination. In *Future information technology* (pp. 403–408). Springer.
35. Sirohey, S., Rosenfeld, A., & Duric, Z. (2002). A method of detecting and tracking irises and eyelids in video. *Pattern Recognition*, *35*(6), 1389–1401.
36. Smith, B., Brandt, J., Lin, Z., & Zhang, L. (2014). Nonparametric context modeling of local appearance for pose-and expression-robust facial landmark localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1741–1748).
37. Song, F., Tan, X., Chen, S., & Zhou, Z.H. (2013). A literature survey on robust and efficient eye localization in real-life scenarios. *Pattern Recognition*, *46*(12), 3157–3173.
38. Song, F., Tan, X., Liu, X., & Chen, S. (2014). Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients. *Pattern Recognition*, *47*(9), 2825–2838.
39. Tan, X., Song, F., Zhou, Z.H., & Chen, S. (2009). Enhanced pictorial structures for precise eye localization under incontrolled conditions. In *IEEE conference on computer vision and pattern recognition, 2009. CVPR 2009* (pp. 1621–1628). IEEE.
40. Tan, X., & Triggs, B. (2010). Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Transactions on Image Processing*, *19*(6), 1635–1650.
41. Turtinen, M., Pietikäinen, M., Turtinen, M., & Pietikäinen, M. (2006). Contextual analysis of textured scene images. In *Proceedings of the 17th British machine vision conference (BMVC 2006)* (Vol. 2, pp. 849–858). Edinburgh, UK.
42. Vapnik, V.N., & Vapnik, V. (1998). *Statistical learning theory* (Vol. 1). New York: Wiley.
43. Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition, 2001* (Vol. 1, pp. I–511). IEEE.
44. Wang, Q., & Yang, J. (2006). Eye detection in facial images with unconstrained background. *Journal of Pattern Recognition Research*, *1*(1), 55–62.
45. Wolf, L., Hassner, T., & Taigman, Y. (2008). Descriptor based methods in the wild. In *Workshop on faces in 'real-Life' images: detection, alignment, and recognition*.
46. Wu, J., & Mei, L. (2013). A face recognition algorithm based on asm and gabor features of key points. In *2012 international conference on graphic and image processing* (pp. 87,686L–87,686L). International Society for Optics and Photonics.
47. Wu, Y., & Ji, Q. (2014). Learning the deep features for eye detection in uncontrolled conditions. In *22nd international conference on pattern recognition (ICPR) 2014* (pp. 455–459). IEEE.
48. Wu, Y.S., Lee, T.W., Wu, Q.Z., & Liu, H.S. (2010). An eye state recognition method for drowsiness detection. In *Vehicular technology conference (VTC 2010-Spring), 2010 IEEE 71st* (pp. 1–5). IEEE.
49. Xiang, C., Ding, S.Q., & Lee, T.H. (2005). Geometrical interpretation and architecture selection of mlp. *IEEE Transactions on Neural Networks/a Publication of the IEEE Neural Networks Council*, *16*(1), 84–96.
50. Xu, C., Zheng, Y., & Wang, Z. (2008). Eye states detection by boosting local binary pattern histogram features. In *15th IEEE international conference on image processing, 2008. ICIP 2008* (pp. 1480–1483). IEEE.
51. Yi, D., Lei, Z., & Li, S.Z. (2011). A robust eye localization method for low quality face images. In *International joint conference on biometrics (IJCB), 2011* (pp. 1–6). IEEE.
52. Zhang, B., Shan, S., Chen, X., & Gao, W. (2007). Histogram of gabor phase patterns (hgpp): a novel object representation approach for face recognition. *IEEE Transactions on Image Processing*, *16*(1), 57–68.
53. Zhao, G., Ahonen, T., Matas, J., & Pietikainen, M. (2012). Rotation-invariant image and video description with local binary pattern features. *IEEE Transactions on Image Processing*, *21*(4), 1465–1477.
54. Zhou, M., Wang, X., Wang, H., Heo, J., & Nam, D. (2015). Precise eye localization with improved sdm. In *IEEE international conference on image processing (ICIP), 2015* (pp. 4466–4470). IEEE.
55. Zhou, Z.H., & Geng, X. (2004). Projection functions for eye detection. *Pattern Recognition*, *37*(5), 1049–1056.

Benrachou Djamel Eddine is a PhD student at Laboratoire d'Automatique et Signaux de Annaba, (LASA), faculty of engineering science Badji Mokhtar Annaba (UBMA), Annaba, Algeria. He received his bachelor's degree in Automatic (2008) and his academic master's degree (2010) in Automatic and Signals from the faculty of Engineering Science Badji Mokhtar Annaba, (UBMA), Annaba, Algeria. His interests lie in Automatic Driver's Drowsiness Monitoring Systems, Human-Machine Interaction, Computer Vision, Image Processing, Deep Learning, and Pattern Recognition.



Filipe Neves dos Santos was born in Santa Maria da Feira, Portugal, in October, 1979. He holds a Master's degree (2007) in Automation and Robotics and a PhD degree (2014) in Electrical and Computer Engineering. He is currently a Senior Researcher at the Technology and Science Associate Laboratory (INESC-TEC), sub-unit CROB (Center of Robotics and Intelligent Systems). His research focuses on agricultural robotics, visual perception, navigation systems, and sensor fusion and integration.

Brahim Boulebtateche received his five-year State Engineer Degree (Diplôme d' Ingénieur d'Etat) with honors from National Polytechnic School of Algiers, Algeria (1984), his MPhil from Bradford University, Bradford, England, in 1988, and his PhD from University of Badji Mokhtar Annaba (UBMA), Annaba, Algeria, in 2007, all in electrical and electronic engineering. Since 1988, he has been with Badji Mokhtar University, Annaba (UBMA), where he is currently teaching in the Department of Electronics and he is a member of Laboratoire d'Automatique et Signaux de Annaba, (LASA). His current research interests include Computer Vision, Machine Learning, Robotics and Intelligent Control Systems.

Salah Bensaoula received his Engineer Degree of State from National Polytechnique of Algiers in 1983. The DEA degree from Clermont-Ferrand, France, and the Doctorate Degree from Saint-Etienne University, France, in 1984 and 1987, respectively. His main interests of research include Fault Detection and Isolation in industrial systems and Man-Machine Communication.