

# Fast Eye Centre Localization Using Combined Unsupervised Technics

Saliha Berrached<sup>1</sup>, Nasr-Eddine Berrached<sup>1</sup>

**Abstract:** Eye movements offer precious information about persons' state. Video surveillance, marketing, driver fatigue as well as medical diagnosis assistance applications manage eye behavior. We propose a new method for efficiently detecting eye movement. In this paper, we combine circle eye model with eye feature method to improve the accuracy. A set of detectors estimate the eyes centers to increase the localization rate. As a pre-processing stage, the mean of the edges yields the center of the two eye regions. Image treatment operations reduce the ROI. A Circle Hough Transform (CHT) algorithm is adopted in a modified version as a detector to find the circle eye in the image; the circle center found represents the eye's pupil estimation. We introduced the Maximally Stable Extremal Region (MSER) as a second detector, which has never been used for eye localization. Invariant to continuous geometric transformations and affine intensity changes and detected at several scales, MSERs efficiently detect regions of interest, in our case eye regions, and precisely, their centers. Ellipses fit MSERs, and their centroid estimation match eyes center. We demonstrate that the true eye centers can be found by combining these methods. The validation of the proposed method is performed on a very challenging BioID base. The proposed approach compares well with existing state-of-the-art techniques and achieves an accuracy of 82.53% on the BioID database when the normalized error is less than 0.05, without prior knowledge or any learning model.

**Keywords:** Centre eye localization, CHT, gaze, MSER, Unsupervised technics.

## 1 Introduction

Automatic human eye tracking and localization is an extensive investigation in computer science technologies. Many human-computer interface devices use eye localization technology. Due to their daily use, Human-Computer Interface (HCI) applications seem to be the most interesting. Eye tracking and detection are focused on two priority areas, eye localization and gaze estimation. It should be noted that there are three aspects related to the first one.

---

<sup>1</sup>Intelligent Systems Research Laboratory, Electronics Department, University of Sciences and Technology of Oran USTO-MB, Algeria; E-mails: [saliha.berrached@univ-usto.dz](mailto:saliha.berrached@univ-usto.dz); [nasreddine.berrached@univ-usto.dz](mailto:nasreddine.berrached@univ-usto.dz)

The primary aspect is to detect the eye existence, the second one is to localize eye position in the image, and the third one is to track the previously located eyes [1]. On computers, smartphones, and other new vision technologies, gaze can offer a comfortable manner for guiding by gaze directions (robotic control) [2]. Eyes are becoming an interesting communication channel between contributors. Other control systems and security applications gain robustness by adding eyes detection and eye center localization: precise eye localization permits suspicious behavior detection in stadiums, airports, ports, theatres and other sites. Person identification improves due to image processing advances.

Also, other commercial sectors benefit from the regarded tracking process; for example, potential customer gaze direction analysis can help to choose most relevant product place. Eye movement is becoming more and more used in medical field to get earlier and more precise diagnoses such as some pathologies related to brain state. All these recent applications prove eyes localization importance focusing on eye center localization.

Different eyes localization methods are summarized in [3], those methods can be divided into two different categories according to acquisition sources. The first category are images obtained under Infra-Red (IR) illumination acquisition, used to yield corneal reflection, which is very helpful for center eye's location. The second category takes into account light variations in a set of images taken from a video stream, this type is closer to real scenarios, however, the extreme variability of eyes' appearance in a real environment makes eyes center detection a very complex challenge. Different approaches are involved in this category of acquisition type. Many challenging difficulties have to be handled as human eye is a very complex organ, its nature varies from a person to another, it has intrinsic changes (open or closed eyes, partially or totally), wearing glasses affects eye image, presence of light modulates pupil dilatation, and eye does not have a simple known shape. All these characteristics necessitate a development of a robust application which can achieve the majority of the requirements. In their paper, Pavlović et al. [4] made the comparison between two imagery types, the visible and the infrared light, where the original size image was 320 x 240 pixels for face identification purpose, and proposed an optimal ratio between cell and image size for calculating Histogram of Oriented Gradients (HOG) features. Results are compared with the original size facial image and conclusion was that downsized dimension images with scale factor from 0.1 to 1 offer better recognition compared with original size images precisely for scale factor equal to 0.2.

Precise eye center location methods differ by computation, efficiency and more over precision degree. The precision degree is typically defined by the relative error measure proposed by Jesorsky et al. [5], which is considered nowadays the major evaluation metric used. Yang et al. [6] proposed a novel Gabor Eye based method that makes full use of the special grey distribution in

the eye-and-brow region, and self adaptively selects proper Gabor kernel to convolute with the face image. The proposed method is robust against the face changes in illumination, expression and pose.

Hamouz et al. [7] develop a new method to localize faces for person identification by ten feature points on face through Gabor filters yielding a superior performance compared with the reference methods. S. Kim et al. [8] use multiscale Gabor Feature vectors at eye coordinates which allowed efficient eye localization. In [9] and [10], authors introduce isophote curvatures to infer the center of (semi) circular patterns and a novel center voting mechanism, whereas in [11] use of isophote curvatures in fatigue detection process provided a detection rate exceeding 85%. Eye's isophote curvature combined to shape regression model is used in [12] to obtain robustness in eye center localization. In [13] isophote curvature is combined with quasi-continuous responses of a modified cascade classifier framework using appearance-based features to study its performance within real scenarios. Timm and Barth [14] proposed to localize eye centers by managing image gradients, an objective function expressed by a periocular geometry that peaked at the center of a circular object.

Efficient SVM, PCA, neural networks, and Fisher linear discriminant classifiers are widely used to locate eyes, respectively, in [15–17] and [5].

Naseem et al. [18] adopt a faster RCNN deep learning model and AlexNet as the detection key for face, eyes and eye openness detection. The localization step combines techniques composed of rectangular-intensity-gradient approach. Levinstein et al. [19] created a hand-crafted data set adopted to train a cascade of regression forests, the obtained results are equivalent to automatically trained systems, the final results are refined by robust circle fitting followed by circle matching process, the most relevant candidate is assumed to be the iris and its center is declared the eye center. Struc and co-authors in [20] introduced a new technique to extract facial landmarks, based on Principal directions of Synthetic Extract Filters (PSEFs) and then applied this method for eye localization. The results were in accordance with the Haar filter method proposed in [21]. Wei-Yen Hsu et al. [22] carried out a new center localization method to locate the eye center under different situations in a large yaw head rotation (between  $-67.5^\circ$  and  $+67.5^\circ$ ).

Fusion and combination of multiple unsupervised techniques is largely used for iris localization. Xiao et al. [23] fused the facial landmarks, active contours (snakuscle), circle fitting, and a simple binary connected component. E. Skodras and N. Fakotakis [24] adopted color information to build an eye map, a cumulative result of the radial symmetry transform is applied both to the eye map and to the original eye image. In [25] M. Soltany et al. proposed grey projection and Circular Hough Transform (CHT) to locate the pupil in natural light eye images.

Leo et al. [26] proposed a new method to locate the eyes, in particular to localize the pupils, by a two-step procedure in which differential analysis of image intensities is combined with self-similarity coefficients, and junction results give the estimated center eye position.

W. Zhang et al. [27] used isophote global center voting and gradient-based pupil estimation in a modular, unsupervised approach. The previous author's papers presented in [23–27] prove the efficiency of eye center localization by unsupervised techniques.

From the above literature, the majority of prior work achieve eye center localization in a known environment using a learning process, which consumes much time and is inappropriate in real-life scenarios. We choose to create algorithm by trying to attempt real-time execution without any previous knowledge about the environment. In this paper, we are proposing a new hybrid method that can achieve robust eye center localization; the main contribution of this research is the combination of several techniques based on mean of edges combined with Circular Hough Transform (CHT) and Maximally Stable Extremal Regions (MSER).

The primary contributions of this research are as follows:

- (i) Proposal of fast approach that combines several methods.
- (ii) Simple techniques are used to create a primary sub window.
- (iii) The MSER technique is applied to the eyes' localization systems. Furthermore, we evaluate robustness by using the very challenging BioID database.

## **2 Eye Center Localization Using an Unsupervised Modular Approach**

The overview of the proposed system is illustrated in Fig. 1. The Viola and Jones algorithm [20] is used to define both the face and eye region in the image. Boosted cascade detectors for rapid object detection based on Haar features are one of the most influential algorithms in object and, in particular, face detection. To extract useful information for our purposes, we must first take an important step in the pre-processing stage. A sequence of image processing operations is performed, which results in a drastic reduction of the region of interest with an enhancement of the essential data. A first necessary estimate of the eye regions and their centers makes use of Viola and Jones algorithm. The sub-windows of detected regions are used by the rest of the elaborated algorithm. The estimated centers provide a very good initial estimate in addition to being used in the case of divergences later on.

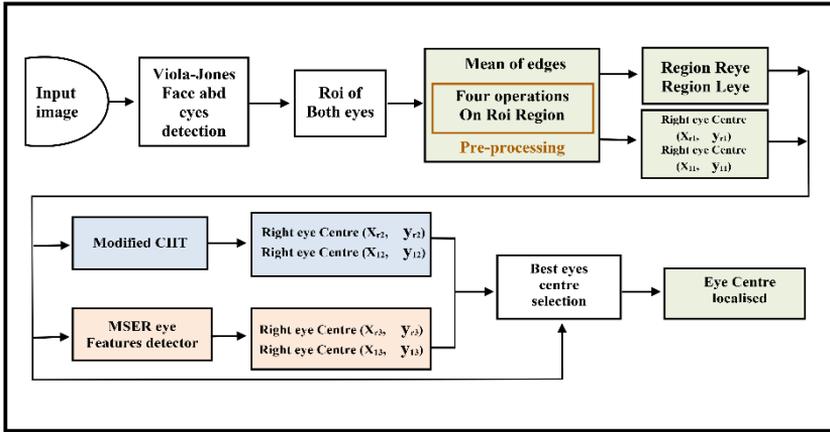


Fig. 1 – An overview of the proposed system.

### 2.1 Eyes separation

The sub-window representing the eye region is divided into two parts, one for each eye (see Fig. 2). The search for the center will be done in these two regions. Governing equations are defined as follows:

$$I(eye) = [i \dots n, j \dots m], \quad (1)$$

obtained from Viola and Jones detector,

$$I(eye)_{left} = [i \dots \text{floor}(n/2), j \dots m], \quad (2)$$

$$I(eye)_{right} = [i + \text{floor}(n/2) \dots n, j \dots m], \quad (3)$$

where  $I(eye)$  represents the eye ROI with size  $(m - i) \times (m - j)$  pixels, (see Fig. 1 (Obtained ROI) starting from the face ROI (Fig. 2b), results of eye separation are shown in Fig. 2c.

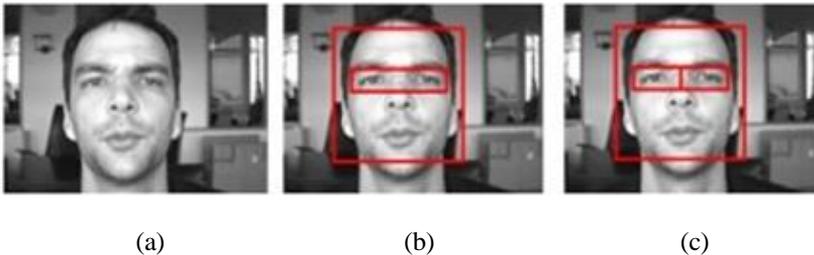


Fig. 2 – Face and eyes detection using Viola and Jones detector [21]:  
 (a) Original image; (b) Face and eyes; (c) Eyes separation.

## 2.2 Image pre-processing

The pre-processing stage in the field of vision and imaging is an operation or a set of operations that make raw data suitable for a thematic analysis. These operations essentially consist of carrying out formatting, edging, geometric and radiometric corrections, and sampling. We applied a series of operations to achieve our objective, see Figs. 3 and 4.

### 2.2.1 Image smoothing

We use average filtering to smoothen our image. The method of spatial domain processing called average (mean) filtering has as its principle the scanning of the entire image by performing a convolution of the image to be processed with a template, in which the value of the scanned pixel is replaced by the average value of the template. Let  $I(x, y)$  be the image to be processed, and  $G(x, y)$  the resulting image. The value of each pixel in the obtained image  $G$  is the average value of the pixel's neighborhood. That can be expressed by the (4).

$$G(x, y) = \sum \frac{f(i, j)}{nb}, \quad i, j \in S, \quad (4)$$

where  $x, y = 1, 2, \dots, nb$  is the number of pixels in the neighborhood,  $nb$  represents a total number of pixels in a template and  $S$  represents the coordinates of each pixel in the eye region. When this operation is applied, it leads to a change in the values of all pixels, which has the effect of making the image blurry with edge effects, so it is imperative to weigh pixels in the template kernel [28], using an example smoothing operator as shown in (5)

$$F = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}. \quad (5)$$

### 2.2.2 Binarization

Based on raw thresholding, we assign a value one (the color white) to a pixel of the image if it has an intensity lower than the threshold, otherwise, it will be zero (black). This process is performed on each pixel of the image, and as a result, we obtain an image including only two levels (values 0 or 1).

### 2.2.3 Closing

This mathematical morphology is applied to remove all undesired blobs around eye region [27].

The closing of  $I$  by a structuring element  $B$  (where  $B$  is a  $3 \times 3$  pixels) is denoted as  $\circ$ , and is obtained by the dilation of  $I$  by  $B$ , followed by erosion of the resulting structure by  $B$ :

$$I \circ B = (I \oplus B_{rot}) \ominus B_{rot}. \quad (6)$$

The dilation of  $I$  by the structuring element  $B$  is denoted by:  $I \oplus B_{rot}$ . The erosion of the binary image  $I$  by the structuring element  $B$  is denoted by:  $I \ominus B$ .

### 2.2.4 Edge extraction

To remove interior pixels, we apply edge extraction method that makes use of neighboring connection, we set a pixel to 0 if all its 4-connected neighbors ((3×3) pixels) are 1, thus leaving only the boundary pixels on.

### 2.2.5 Cropping

We select the area around the edge extracted region of the image. This area represents the region of interest, and the center of these two regions (for each eye) forms a first estimate of the left and right eye centers.



Fig. 3 – Pre-processing steps.

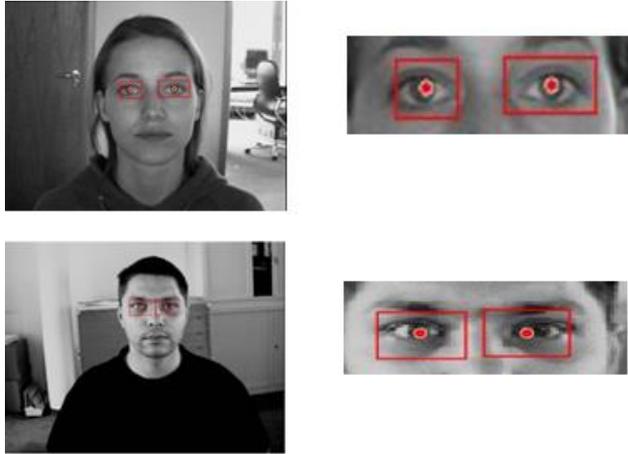


Fig. 4 – Pre-processing results ( $I(eye)_{left1}, C(eye)_{left1}, I(eye)_{right1}, C(eye)_{right1}$ ).

## 3 CHT Detector

We use a Circular Hough Transform (CHT) based algorithm to find the eye circle in the image. This approach is used due to its robustness under different illumination and noise conditions, which is in line with our requirements. Due to the possibility of circle parameters being directly transferred to parameters space, circle is simpler to represent compared to the line [29]. For the CHT calculation,

a separate circle filter can be used for each circle radius to be detected. This forms the familiar 3-dimensional parameter space, usually associated with the CHT, where two dimensions represent the position of the circle center ( $a, b$ ), and the third is its radius  $r$ ,

$$r^2 = (x - a)^2 + (y - b)^2. \quad (7)$$

In parametric form, a circle is expressed by (8) and (9),

$$X = a + r \cos \theta, \quad (8)$$

$$Y = b + r \sin \theta. \quad (9)$$

Therefore (from Fig. 5) in the  $(a, b)$  space, a circle with a radius  $r$  and a centre  $(x, y)$  is represented as follows:

$$a = x - r \cos \theta, \quad (10)$$

$$b = y - r \sin \theta, \quad (11)$$

$(a, b)$  represent polar coordinate for center  $(x, y)$  converted to radians.

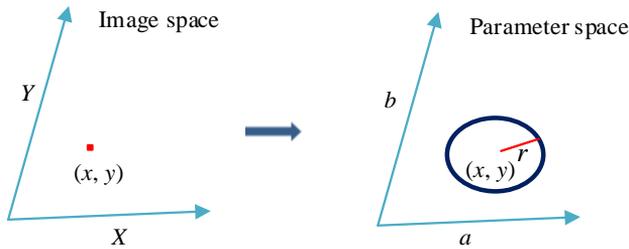


Fig. 5 – From Image to parameter space.

The CHT pseudo-algorithm shows very explicitly CHT limitations.

CHT classical pseudo-Algorithm:

```

-  $H[a, b, r] = [0; 0; 0]$ ; % Initialization of  $H[a, b, r]$ 
-  $I(x, y)$  = image Edge; % the image edges are taken
-Do
BEGIN
    For  $r := 0$  to  $d$  % where  $d$ =diagonal image length
        For  $(x := i$  to  $n)$  and  $(y := j$  to  $m)$ 
            For  $(\theta := 0$  to  $360)$ 
                 $a = x - r \cos \theta$ ;
                 $b = y - r \sin \theta$ ;
                 $H[a, b, r] = H[a, b, r] + 1$ ;
            Until  $H[a, b, r] > t$ ; %  $t$ =threshold value;
        END
    END
END

```

We must deal with the cells fielding accumulator problem (for each accumulator we instantiate 3D matrix), in addition to the 3D parameter space. CHT classical algorithm is computationally very intensive and takes considerable time. Votes for several radii must be stored in a 3D array when using the conventional Hough Transform, which increases storage needs and processing times. For these reasons, the algorithm implementation is modified to be efficient and fast.

### 3.1 Circle Hough Transform Modification

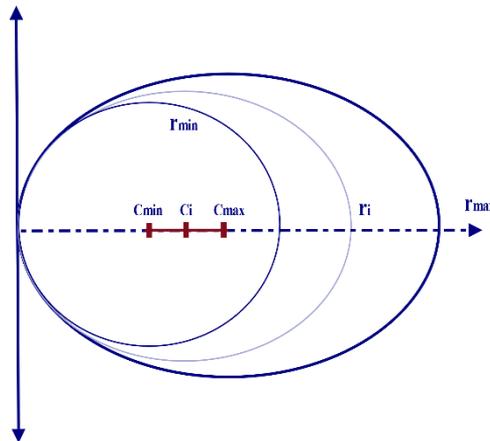
Many different approaches can be taken for the CHT implementation. In this study, we modified the algorithm in order to lower the computation time. There are three important steps that are common to all approaches: accumulator array computation, centre estimation, and radius estimation.

We used the following parameters in the CHT implementation:

Instead of using an accumulator array for each radii, a single 2-D accumulator covers all radii, by using this strategy the overall computational time is lowered, especially when working across a wide radius range. The pre-treatment step significantly reduces calculation here.

Edge Pixels are used because the number quantity of candidate pixels substantially influences both overall memory needs and performance. The input image's gradient magnitude is thresholded to ensure that only pixels with a high gradient are counted in order to reduce their number.

Edge Orientation Information, restricting the number of cells available to candidate pixels improves performance. For these aims, voting process is allowed for a short period along the gradient. The radius range defined by  $r_{\min}$  and  $r_{\max}$  determines the width of the voting interval between points  $c_{\min}$  and  $c_{\max}$ , Fig. 6.



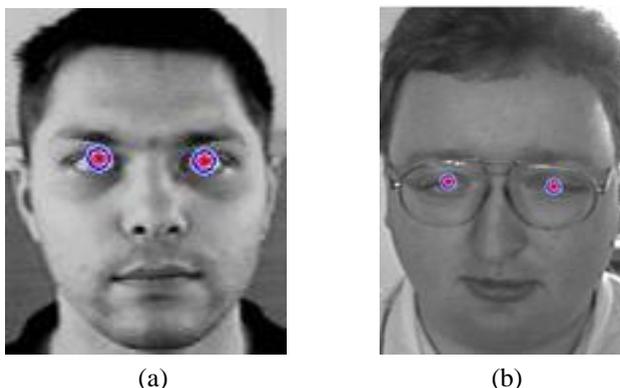
**Fig. 6** – Width of the voting interval between points.

The modified Hough Transform algorithm is used to determine the iris circle for a range of values, with the range being narrowed down to [1, 30] through assessment experiments. Iris contour matching is completed by the modified circular Hough Transform technique, Fig. 7.

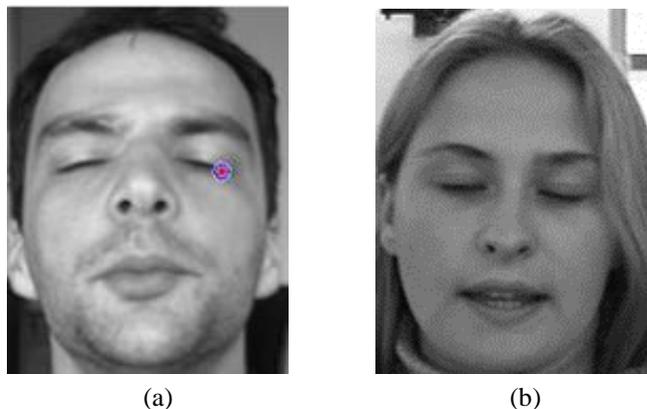
Eyelids, eyelashes, closed or open eyes, and even a person wearing glasses are correctly detected, see examples in Fig. 8. On the other hand, false edges disturb the CHT detector and the method can diverge and provide false detections with no localization or only one eye accurate localization, Fig. 9.



**Fig. 7** – Centre eye localization steps by CHT modified.



**Fig. 8** – Accurate localization by CHT Modified: (a) True localization in both eyes; (b) True localization in both eyes in presence of glasses.



**Fig. 9** – Examples of inaccurate localization by CHT Modified algorithm: (a) No localization of the right eye and false localization of the left eye; (b) No localization of both eyes.

By using a second detector that operates in parallel with the CHT detector, these flaws can be avoided, and the rate of accurate localization will be enhanced. The CHT method is limited by incomplete, insufficient and partial boundary detection. Partially, and completely closed or occluded eyes are not localized correctly.

## 4 MSER Maximally Stable Extremal Region Eye-Features detector

In this paper, we propose Maximally Stable Extremal Region (MSER) concept for eye localization, because of its nature, it is invariant to continuous geometric transformations and affine intensity changes and detected at several scales. MSER is further considered as the fastest interest point detection method. This algorithm was proposed by Matas et al. [30]. It starts from the intensity function of an image and ends with Maximally Stable Extremal Regions, where each region is defined by an extremal property of the intensity function within the region to the values on its outer boundary. Elliptical frames are attached to the MSERs by fitting ellipses to the regions. Those regions' descriptors are kept as features.

### 4.1 MSER eye Features detector

To find stable regions, the MSER detector iteratively scans the input image's intensity range. Regions are extracted from connected components known as "Extremal Regions". The Threshold-Delta parameter controls how many increments are tested for stability by the detector. The MSER object examines changes in region area between various intensity thresholds. To be deemed stable, the variation must be lower than the Max-Area-Variation parameter's value. A local minimum of the relative growth of their square is computed to represent "Maximally Stable" regions [31]; meaning that they have a lower variation than the regions one level below or above. Each Maximally Stable Extremal Region is approximated by an ellipse, those steps are shown in Fig. 10 and in pseudo code MSERs.

The region  $Q(t)$  (where  $t$  denotes its threshold level) is regarded as MSER if the derivative of the region area  $q(t)$  is over the threshold value.  $\text{ThresholdDelta} = t_j - t_{j-1}$ .

$$q(t) = \frac{\frac{d}{dt} \|Q(t)\|}{\|Q(t)\|}. \quad (12)$$

Equation (12) is approximated by:

$$q(t_j) = \frac{\|Q(t_j) - Q(t_{j-1})\|}{\|Q(t_j)\|}, \quad (13)$$

where  $\text{ThresholdDelta} = t_j - t_{j-1}$  defines the threshold increment.

**Pseudo code MSERs:**

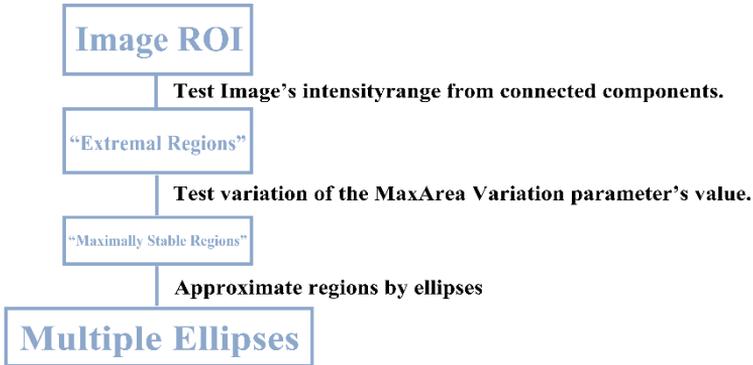
```

MSERegions=detectMser (which are described by pixel lists stored inside the
returned 'regions' object)
Begin
  For 0 < ThresholdDelta < 255 (This value specifies the step size between intensity
  threshold levels used in selecting extremal regions while testing for their stability)
  %Compute a function,  $q(t_j)$ , at each threshold value  $t_j$ 

$$q(t_j) = \frac{\|Q(t_j) - Q(t_{j-1})\|}{\|Q(t_j)\|}$$

  % Analyze this function for each potential region to determine those that have
  similar function value over multiple thresholds.)
  % Update Map of regions by new  $Q(t_j), Q(t_{j-1})$ 
  % Approximate a region with an ellipse
  End.

```



**Fig. 10 – MSERs extraction steps.**

We apply MSER algorithm to the regions  $R_{eye}$  and  $L_{eye}$  (calculated in pre-processing step) according to the minima previously discovered, regions in each blob that correspond to those minima are considered as MSERs with the constraint of additional requirements that define the allowable minimum and maximum MSER region sizes. These requirements are designated by:

- Region Area Range;
- Threshold Delta;
- Max area variation (typical values range from 0.1 to 1.0);
- ROI =  $R_{eye}$  and  $L_{eye}$ .

## 4.2 Restricting the number of regions

A large number of regions are created by the aforementioned technique, and these areas are pruned by ensuring that: Threshold-Delta, Region-Area and Max-Area-Variation are above corresponding thresholds 1.5, between [100,180], and 0.25 (see pseudo code of MSERs selection). All those limitations in the MSER code eliminate regions that do not correspond to the eye region. The results of the method for the maximally stable eye region (MSER) are shown in Fig. 11.

### Pseudo code MSERs Selection”

```

Begin
For i=1 to n (n represents MSERs found)
  Region(i)=false; (initialization)
  If (Region(i).RegionArea ∈ [100 180]) and (Region(i).ThresholdDelta ≥ 1.5 )
and (Region(i).MaxVariation ≥ 0.25 )
    Region(i)=True; (Region(i) is selected )
End.
```

To choose the best region which represents real eye, we introduce pseudo code MSERs Selection algorithm, where n represents number of ellipses found and Max-Variation is defined by iris nature. Our algorithm defines which Mser-Feature from multiple Mser-Features is the most relevant to be the real eye region. Centre of each eye is centre of ellipses obtained by MSER localization.

The equation's best-fit ellipses (given by (14)) ultimately come close to the positions of the discovered MSERs, resulting in a single region that represents the eye region by an ellipse and the centre by an ellipse centroid.

The ellipses equation is given by:

$$a(x - x_0)^2 + 2b(x - x_0)(y - y_0) + c(y - y_0)^2 = 1, \quad (14)$$

where  $(x_0, y_0)$  is the centroid's region centre of the mass computed by calculating moments of order 0 and 1, and  $(a, b, c)$  are coefficients for which it holds  $a^2 - b^2 = c^2$ , and which are calculated from the center moment of order 2. The computational expenses are limited (mostly the conversion of region moments into ellipse parameters), since moments (of any order) for all areas of a segmented image may be precomputed in a single scan of the image.

## 4.3 Discussion

The obtained results demonstrate the applicability of MSER to eye detection on faces under different conditions including wearing glasses, and partially or completely closed eyes (see Fig. 11). Our approach is based on connectivity within the region and its stability and it performs relatively accurate and precise in the localization of eye centers. When MSER regions are localized, these regions can be tracked over time from frame to frame by the MSER algorithm. MSER algorithm is adapted for real time applications as in [32].



Fig. 11 – Accurate localization by MSER.

## 5 Best Eyes Centre Selection

The discovered eye centre candidates by CHT modified algorithm (center of the circle), MSER (gravity center of the ellipse), and even by the mean of edges (pre-processing steps) are sorted according to their centre scores to obtain the most relevant centres. Different criteria used for the scores are as in [33]:

- Right and left eye centres are aligned within range of  $[-30^\circ, 30^\circ]$ ;
- The distance between the two eyes is in the range of  $[1/5, 4/5]$  of the face size;
- In cases of identical scores, an advantage is given to the CHT centers;
- If the CHT and MSER methods do not lead to a result, the centres obtained by the pre-processing represent the final result. Thus, our approach gives results while the face is previously detected (Fig. 12).

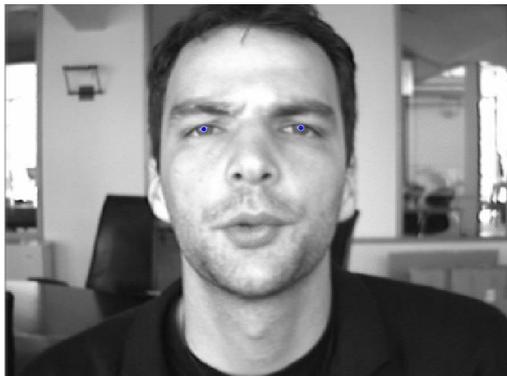


Fig. 12 – Best eyes centers selection.

## 6 Evaluation

Bio ID database [33] is chosen to test our algorithm, because of two major reasons: it is one of the most used datasets in our domain hence it enables comparison with the recent state of the art work. In addition, it is rich in information encompassing differences in subjects, pose, environment and even illumination conditions. The database consists of 1521 grey level images of 23 different subjects. The image quality and the image size (286x384) is approximately equal to the quality of a low-resolution webcam. The left and right eye centers are annotated and provided together with the images. Our approach starts with face and eye region position offered by Viola and Jones detection algorithm, eye centres are estimated on two regions obtained by several steps (as explained in Section 2) based on gradient MSERs and CHT were combined to calculate accurate eye centers. We evaluate the normalized error, which indicates the error obtained by the worst of both eye estimations.

This measure was introduced by Jesorsky et al [5], and is defined as:

$$e \leq \frac{1}{d} \max(e_l, e_r), \quad (15)$$

where  $e_l, e_r$  are the Euclidean distances between the estimated and the correct left and right eye centers, and  $d$  is the distance between the eye centres.

Other metric values  $e_{Average}$ ,  $e_{Best}$  were largely used to demonstrate accuracy of the proposed algorithms: these two normalized errors indicate in a large area representing successively eye region  $e \leq 0.25$ , iris regions  $e \leq 0.10$  and pupil  $e \leq 0.05$  how close is the found location to the real eye center. We also provide the measures  $e_{Best}$  and  $e_{Average}$  in order to observe smallest error as well as an averaged error.

$$e_{Average} \leq \frac{1}{2d} \max(e_l, e_r), \quad (16)$$

$$e_{Best} \leq \frac{1}{d} \min(e_l, e_r). \quad (17)$$

## 7 Results

The evaluation of the proposed algorithm yields precise localization of eye pupil, as shown in Fig. 13. **Table 1** contains the obtained results to verify the method's effectiveness. Note that our algorithm consists of four levels:

- The mean of the gradients to achieve a starting step for subsequent treatments. It reduces ROI region and yields an accuracy of 64.65% for  $e \leq 0.25$  (eye region);

- MSERs are effective in dealing with eyes that are partially or even completely closed and provide 89.08% for  $e \leq 0.25$  ;
- The CHT approach is powerful in different scenarios because of the nature of the eye (when eyes shape in the image approximates a circle shape, eyes open) with effectiveness of 91.78% for  $e \leq 0.25$  ;
- The combined approach produces the best accuracy of 98.85% for  $e \leq 0.25$  and we obtain 82.53% for  $e \leq 0.05$  (pupil localization).

**Table 1**

*Accuracy vs. normalized error for applying method.*

METHODS	$e \leq 0.05$	$e \leq 0.10$	$e \leq 0.25$
MEAN OF GRADIENTS	20.37%	58.62%	64.65%
MSER	45.21%	64.44%	89.08%
CHT	58.62	72.66%	91.78%
MEAN OF GRADIENTS + MSERS + CHT	82.53%	89.81%	98.85%

From the accuracy curve of the obtained results for different normalized error  $e$  (relative error measure proposed by Jesorsky and al. [5]) (red line, Fig. 13), the y-axis reports the percentage of images in the database on which the pupils were localized in terms of the error parameter  $e$  calculated by (15), x-axis reports the corresponding value of  $e$  (error less than the normalized error). The same figure also reports the pupil localization performances obtained on the same database by using two other metrics,  $e_{Average}$  (black line calculated by (16)) and  $e_{Best}$  (blue line calculated by (17)). For  $e \leq 0.05$ , 10 and 0.25, the proposed method obtains accuracy results of 82.53, 89.81 and 98.85% respectively on the BioId database.

**Table 2**

*Comparison of the accuracy for eye center localization on the BioID dataset.*

Method	$e \leq 0.05$	$e \leq 0.10$	$e \leq 0.25$
Jesorsky et al. (2001) [5]	38.0%	78.8%	91.8%
Timm and Barth (2011) [14]	82.5%	93.4%	98.0%
Valenti and Gevers (2012) [10]	86.09%	91.67%	97.87%
Leo M et al. (2014) [26]	80.67%	87.31%	94%
Weng Zeng et al. (2016) [27]	85.66%	93.68%	99.21%
Wei-Yen Hsu et al. (2021) [22]	75.79%	94.54%	99.61%
Proposed Method	82.53%	89.81%	98.85%

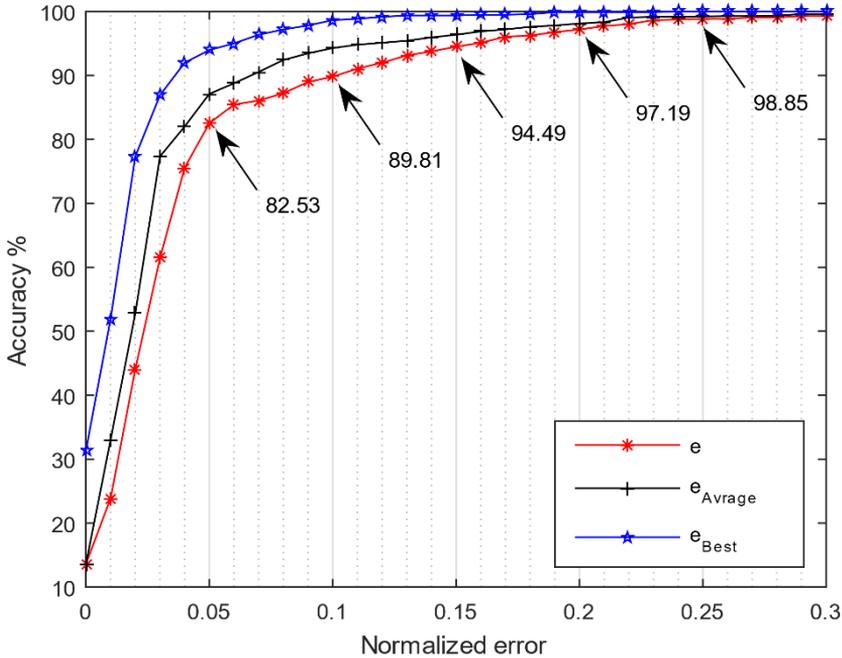


Fig. 13 – Accuracy curve of the proposed method on the BioID dataset.

## 8 Comparison with State of the Art and Discussion

This paper presents a new study for accurate eye center localization, which is compared with the existing results from other top state-of-the-art methods in literature. According to **Table 2**, we obtain 82.85% for  $e \leq 0.05$  which is similar to other unsupervised technics. For  $e \leq 0.25$  we achieve 98.85% accuracy, these results are very much in line with the study conducted by [14, 22, 27].

Our results are very encouraging, the proposed strategy provides similar results with the majority of the relevant approaches, including some that make use of supervised training or post-processing adaptations and also, our algorithm delivered promising results in that regard. Our strategy is appropriate for application with eye center localization such as fatigue and driver attention detection, gaze tracking and even real time eye localization.

### 8.1 Processing time

The proposed method is simulated by the MATLAB (R2015a) numeric computing platform and executed on an Asus PC with 8 GB/RAM and a 1.8 GHz Intel i7-4500U CPU processor installed with Windows 8.1. For the experiment,

average processing time is calculated using the BioID database. The size of each image is  $384 \times 288$  pixels.

**Table 3** compares the average processing time of several algorithms that enable near real time performance, taking both hard and soft implementation into account. The proposed method delivers lower processing time compared to others results. The MSER eye center localization process alone takes 0.04 s. The processing time will be shorter by applying MSER for tracking the eye center between successive images and not for detection on each image separately, and this has enabled us to improve our results.

**Table 3**  
*Processing time for most common algorithms for the eye localization.*

Methods	Implementation	Average processing time
Proposed method	Matlab (R2015a) platform and is tested on Asus computer with 8 GB RAM and 1.8 GHz Intel i7-4500U CPU processor	0.085 s
Vater, Sebastian et León (2016) [12]	Standard CPU with 3.3GHZ in a prototype Matlab implementation	0.1 s
Chen, Shuo et LIU, Chengjun (2015) [14]	Pentium 3with 3.0 GHZ	0.12 s
Levinshtein et al. (2018) [18]	Modern Laptop Xeon 2.8GHZ CPU	0.04 s (not including the face detection)
Xiao, Fet al. (2018) [22]	Laptop with an intel ® Core ™ i7, 7500U CPU running at 2.7GHZ an 8GB RAM	0.0257 s
Leo M, Cazzato (2014) [25]	Matlab (R2012a) platform and is tested on Sony VAIO PGG_71213w processor	0.07 s
Ahmed, Manir et al. (2022) [32]	Matlab (R2017b) platform and is tested on Dell computer with 16 GB RAM and 3.30 GHz Intel Core i5 CPU processor	0.0657 s

## 9 Conclusion

Eye localization method based on an unsupervised system made of multiple combined techniques is proposed, which employs the combined CHT and MSER algorithm to achieve robust and fast eye localization. The objectives of strength, ease, and inexpensive imaging devices were achieved with the proposed algorithm. Introducing CHT modified method enables good results on images under complicated conditions (pose, luminance, etc.). The MSER approach improves eye center estimation by resolving partially or totally closed eyes due to connecting blobs technique. In particular, our algorithm achieves an accuracy of 80.53% with an error less than the normalized error 0.05.

We summarize our contributions to three major points:

- A pre-processing step (first detector) that achieves first estimation.
- Reduce noises and offer excellent area of interest, in particular minimize processing time and increase accuracy rate by introducing fast technics such as MSER.
- A good choice of unsupervised collaborative technics that assume constraints linked to HCI application.
- Introduction of MSER eye features invariant to continuous transformation and affine intensity changes is very suitable for eye center localization.

## 10 References

- [1] D.W. Hansen, Q. Ji: In the Eye of the Beholder: A Survey of Models for Eyes and Gaze, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 32, No. 3, March 2010, pp. 478–500.
- [2] L. Itti, C. Koch: Computational Modelling of Visual Attention, Nature Reviews Neuroscience, Vol. 2, No. 3, March 2001, pp.194–203.
- [3] F. Song, X. Tan, S. Chen, Z.- H. Zhou: A Literature Survey on Robust and Efficient Eye Localization in Real-Life Scenarios, Pattern Recognition, Vol. 46, No. 12, December 2013, pp. 3157–3173.
- [4] M. Pavlović, B. Stojanović, R. Petrović, S. Puzović, S. Stanković: Optimal HOG Cell to Image Ratio for Robust Multi-Sensor Face Recognition Systems, Serbian Journal of Electrical Engineering, Vol. 16, No. 3, October 2019, 387–403.
- [5] O. Jesorsky, K. J. Kirchberg, R. W. Frischholz: Robust Face Detection Using the Hausdorff Distance, Proceedings of the International Conference on Audio- and Video-Based Biometric Person Authentication, Halmstad, Sweden, June 2001, pp. 90–95.
- [6] P. Yang, B. Du, S. Shan, W. Gao: A Novel Pupil Localization Method Based on GaborEye Model and Radial Symmetry Operator, Proceedings of the International Conference on Image Processing (ICIP), Singapore, Singapore, October 2004, pp. 67–70
- [7] M. Hamouz, J. Kittler, J. K. Kamarainen, P. Paalanen, H. Kälviäinen, J. Matas: Feature-Based Affine-Invariant Localization of Faces, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 9, September 2005, pp. 1490–1495.
- [8] S. Kim, S.- T. Chung, S. Jung, D. Oh, J. Kim, S. Cho: Multi-Scale Gabor Feature Based Eye Localization, World Academy of Science, Vol. 21, 2007, pp. 483–487.
- [9] R. Valenti, T. Gevers: Accurate Eye Center Location and Tracking Using Isophote Curvature, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, USA, June 2008, pp. 1–8.
- [10] R. Valenti, T. Gevers: Accurate Eye Center Location Through Invariant Isocentric Patterns, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 34, No. 9, September 2012, pp. 1785–1798.
- [11] Z. Deng, R. Jing, L. Jiao, L. Liu: Fatigue Detection Based on Isophote Curve, Proceedings of the International Conference on Computer and Computational Sciences (ICCCS), Greater Noida, India, January 2015, pp. 146–150.

- [12] C. Wei, Z. Pang, D. Chen: Combining Shape Regression Model and Isophotes Curvature Information for Eye Center Localization, Proceedings of the 7<sup>th</sup> International Conference on Biomedical Engineering and Informatics, Dalian, China, October 2014, pp. 156–160.
- [13] S. Vater, F. Puente León: Combining Isophote and Cascade Classifier Information for Precise Pupil Localization, Proceedings of the IEEE International Conference on Image Processing (ICIP), Phoenix, USA, September 2016, pp. 589–593.
- [14] F. Timm, E. Barth: Accurate Eye Centre Localisation by Means of Gradients, Proceedings of the International Conference on Computer Vision Theory and Applications, Vilamoura, Portugal, March 2011, pp. 125–130.
- [15] S. Chen, C. Liu: Eye Detection Using Discriminatory Haar Features and a New Efficient SVM, Image and Vision Computing, Vol. 33, January 2015, pp. 68–77.
- [16] B. Kroon, A. Hanjalic, S. M. P. Maas: Eye Localization for Face Matching: Is it Always Useful and Under What Conditions?, Proceedings of the International Conference on Content-Based Image and Video Retrieval, Niagara Falls, Canada, July 2008, pp. 379–388.
- [17] D. E. Benrachou, F. N. Dos Santos, B. Boulebtateche, S. Bensaoula: Automatic Eye Localization; Multi-Block LBP vs. Pyramidal LBP Three-Levels Image Decomposition for Eye Visual Appearance Description, Proceedings of the 7<sup>th</sup> Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA), Santiago de Compostela, Spain, June 2015, pp. 718–726.
- [18] N. Ahmad, K. Singh Yadav, M. Ahmed, R. H. Laskar, A. Hossain: An Integrated Approach for Eye Centre Localization Using Deep Networks and Rectangular-Intensity-Gradient Technique, Journal of King Saud University-Computer and Information Sciences, Vol. 34, No. 9, October 2022, pp. 7153–7167.
- [19] A. Levinshtein, E. Phung, P. Aarabi: Hybrid Eye Center Localization Using Cascaded Regression and Hand-Crafted Model Fitting, Image and Vision Computing, Vol. 71, March 2018, pp. 17–24.
- [20] V. Štruc, J. Žganec Gros, N. Pavešić: Advanced Correlation Filters for Facial Landmark Localization, Elektrotehniški vestnik, Vol. 79, No. 4, 2012, pp. 209–212.
- [21] P. Viola, M. Jones: Rapid Object Detection Using a Boosted Cascade of Simple Features, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Kauai, USA, December 2001, pp. I-I.
- [22] W.- Y. Hsu, C.- J. Chung: A Novel Eye Center Localization Method for Multiview Faces, Pattern Recognition, Vol. 119, November 2021, p. 108078
- [23] F. Xiao, K. Huang, Y. Qiu, H. Shen: Accurate Iris Center Localization Method Using Facial Landmark, Snakuscule, Circle Fitting and Binary Connected Component, Multimedia Tools and Applications, Vol. 77, No. 19, October 2018, pp. 25333–25353.
- [24] E. Skodras, N. Fakotakis: An Accurate Eye Center Localization Method for Low Resolution Color Imagery, Proceedings of the IEEE 24<sup>th</sup> International Conference on Tools with Artificial Intelligence, Athens, Greece, November 2012, pp. 994–997.
- [25] M. Soltany, S. T. Zadeh, H.- R. Pourreza: Fast and Accurate Pupil Positioning Algorithm Using Circular Hough Transform and Gray Projection, Proceedings of the International Conference on Computer Communication and Management, Singapore, Singapore, January 2011, pp. 556–561.
- [26] M. Leo, D. Cazzato, T. De Marco, C. Distanto: Unsupervised Eye Pupil Localisation through Differential Geometry and Local Self-Similarity Matching, PLOS ONE, Vol. 9, No. 8, August 2014, p. e102829.

- [27] W. Zhang, M. L. Smith, L. N. Smith, A. R. Farooq: Eye Centre Localisation: An Unsupervised Modular Approach, *Sensor Review*, Vol. 36, No. 3, June 2016, pp. 277–286.
- [28] P. Li, H. Wang, M. Yu, Y. Li: Overview of Image Smoothing Algorithms, *Journal of Physics: Conference Series*, Vol. 1883, No. 1, 2021, p. 012024.
- [29] N. Cherabit, F. Z. Chelali, A. Djeradi: Circular Hough Transform for Iris Localization, *Science and Technology*, Vol. 2, No. 5, September 2012, pp. 114–121.
- [30] J. Matas, O. Chum, M. Urban, T. Pajdla: Robust Wide Baseline Stereo from Maximally Stable Extremal Regions, *Proceedings of the British Machine Vision Conference (BMVC)*, Cardiff, UK, September 2002, pp. 384–393.
- [31] A. Sluzek, H. Saleh, B. Mohammad, M. Al-Qutayri, M. Ismail: MSER-in-Chip: An Efficient Vision Tool for IoT Devices, Ch. 14, *The IoT Physical Layer: Design and Implementation*, 1<sup>st</sup> Edition, Springer, Cham, 2019.
- [32] M. Donoser, H. Bischof: Efficient Maximally Stable Extremal Region (MSER) Tracking, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, New York, USA, June 2006, pp. 553–560.
- [33] M. Ahmed, R. H. Laskar: Eye Center Localization Using Gradient and Intensity Information Under Uncontrolled Environment, *Multimedia Tools and Applications*, Vol. 81, No. 5, February 2022, pp. 7145–7168.
- [34] BioID Technology Research, The BioID Face Database, Available at: <https://www.bioid.com/face-database/>