This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

Accurate Eye Center Location through Invariant Isocentric Patterns

Roberto Valenti, *Student Member, IEEE,* and Theo Gevers, *Member, IEEE,*

Abstract—Locating the center of the eyes allows for valuable information to be captured and used in a wide range of applications. Accurate eye center location can be determined using commercial eye-gaze trackers, but additional constraints and expensive hardware make these existing solutions unattractive and impossible to be used on standard (*i.e.* visible wavelenght), low resolution images of eyes. Systems based solely on appearance are proposed in literature, but their accuracy does not allow to accurately locate and distinguish eye centers movements in these low resolution settings. Our aim is to bridge this gap by locating the center of the eye within the area of the pupil on low resolution images, taken from a webcam or a similar device. The proposed method makes use of isophote properties to gain invariance to linear lighting changes (contrast and brightness), to achieve in plane rotational invariance, and to keep low computational costs. To further gain scale invariance, the approach is applied to a scale space pyramid. In this paper, we extensively test our approach for its robustness to changes in illumination, head pose, scale, occlusion and eye rotation . We demonstrate that our system can achieve a significant improvement in accuracy over state of the art techniques for eye center location in standard low resolution imagery.

Index Terms—Eye center location, isophotes, facial features detection.

1 Introduction

As shown by increasing interest on the subject [8], [22], [28], eye center location is an important component in many computer vision applications and research. In fact, the information about the location of the eye center is commonly used in applications as face alignment, face recognition, human-computer interaction, control devices for disabled people, user attention and gaze estimation (*e.g.* driving and marketing) [21], [6]. Eye center location techniques can be divided into three distinct categories which employ different modalities [15]: (1) Electro oculography, which records the electric potential differences of the skin surrounding the ocular cavity; (2) scleral contact lens/search coil, which makes use of a mechanical reference mounted on a contact lens, and (3) photo/video oculography, which uses image processing techniques to locate the center of the eye. The highly accurate eye center information obtained through the mentioned modalities is often used in eye-gaze trackers to map the current position of the eyes to a known plane (*i.e.* a computer screen) as a user's visual gaze estimate. Unfortunately, the common problem of the above techniques is the requirement of intrusive and expensive sensors [4]. In fact, while photo/video oculography is considered the least invasive of the described modalities, commercially available eye-gaze trackers still require the user to be either equipped with a head mounted device, or to acquire high resolution eye images through zoomed cameras [9] combined with a chinrest to limit the allowed head movement. Furthermore, daylight applications are precluded due to the common use of active infrared (IR) illumination to obtain accurate eye location through corneal reflection [35]. Approaches that fuse IR and appearance based modalities are also proposed in literature [50], but dedicated hardware is still required.

In situations in which a closed up/infrared image of the eye is not available, the low resolution information about the location of the center of the eye is still very useful for a large number of applications (*e.g.* detecting gaze aversion, estimating the area of interest, automatic red eye reduction, landmarking, face alignment, gaming, and HCI). Therefore, in this paper we want to focus on appearance based eye locators which can operate when infrared corneal reflections or high resolution eye images are not available. Many appearance based methods for eye center locators in low resolution settings are already proposed in literature, which can be roughly divided in three methodologies: (1) Model based methods, (2) Feature based methods and (3) Hybrid methods.

The model based methods make use of the holistic appearance of the eye (or even of the face). These approaches often use classification of a set of features or the fitting of a learned model to estimate the location of the eyes (possibly in combination with other facial features). By using the global appearance, model based methods have the advantage of being very robust and accurate in detecting the overall eye locations. However, as the success of these methods depends on the correct location of many features or the convergence of a full model, the importance of eye center location is often neglected due to its variability and learned as being in the middle of the eye model or of the two eye corner features. Therefore, in these cases, since the rest of the model is still correct and the minimization function satisfied, these methods are usually not very accurate when they are faced with subtle eye center movements.

On the contrary, features based methods use well known eye

- *Roberto Valenti and Theo Gevers are with the Intelligent System Lab Amsterdam, University of Amsterdam, Science Park 107, 1098 XG Amsterdam, The Netherlands.*
  E-mail: {r.valenti, th.gevers}@uva.nl
- *This paper updates and extends an earlier publication [43] in CVPR 2008.*

Digital Object Indentifier 10.1109/TPAMI.2011.251

Method	Pre-Requirements	Approach	Uses Learning	Used Feature	Used Model/Learning Scheme
Asteriadis [2]	Detected face	Feature Based	-	Edges	Eye model for init + edge crossing count
Jesorsky [27]	Converged face model	Model Based	X	Edges	Hausdorff distance on eye model
Cristinacce [13]	Detected face	Model Based	X	Pixels	PRFR + AAM
Türkan [42]	Detected face	Hybrid	X	Edges	SVM
Bai [3]	Detected face	Feature Based	-	Gradient	-
Wang [47], [48]	Detected face	Model Based	X	RNDA	Boosted classifiers cascade
Campadelli [7]	Detected face	Hybrid	X	Haar Wavelets	SVM
Hamouz [24]	Correct constellation	Model Based	X	Gabor filters	Constellation of face features + GMM
Kim [30]	Normalized face images	Model Based	X	Gabor jets	Eye model bunch
Niu [36]	Detected face	Model Based	X	Haar Wavelets	Boosted classifiers cascade
Wang [46]	Both eyes visible	Hybrid	X	Topographic labels	SVM
Huang [26]	Detected face	Hybrid	X	Mean, std, entropy	Genetic Algorithms + Decision trees
Reale [38]	Detected face	Model Based	-	Pixels	Circle Fitting
Asadifard [1]	Detected face	Feature Based	-	Pixels	CDF filtering
Timm [41]	Detected face	Feature Based	-	Gradient	-
Kroon [32]	Detected face	Model Based	X	Pixels	Elastic bunch graph + LDA
Our basic method	Detected face	Feature Based	-	Isophotes	-
Our enhanced method	Detected face	Hybrid	X	SIFT	kNN

TABLE 1
Differences between the methods discussed in this paper.

properties (*i.e*. symmetry) to detect candidate eye centers from simple local image features (*e.g*. corners, edges, gradients), without requiring any learning or model fitting. Therefore, when the feature based methods are not confused by noise or surrounding features, the resulting eye location can be very accurate. However, as the detected features might often be wrong, the feature based methods are less stable then the model based ones.

Finally, in the hybrid methods, the multiple candidate eye locations obtained by a feature based method are discriminated by a classification framework, therefore using a previously learned eye model to achieve better accuracy.

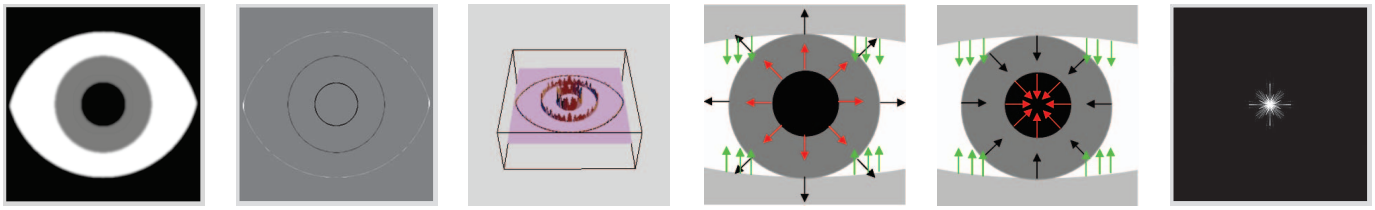
Within the state of the art methods in each of the described methodologies, we studied the following subset: The method proposed by Asteriadis *et al*. [2] assigns a vector to every pixel in the edge map of the eye area, which points to the closest edge pixel. The length and the slope information of these vectors is consequently used to detect and localize the eyes by matching them with a training set. Jesorsky *et al*. [27] proposed a face matching method based on the Hausdorff distance followed by a Multi-Layer Perceptron (MLP) eye finder. Cristinacce *et al*. [13], [12] utilize a multistage approach to detect facial features (among them the eye centers) using a face detector, Pairwise Reinforcement of Feature Responses (PRFR), and a final refinement by using Active Appearance Model (AAM) [11]. Türkan *et al*. [42] apply edge projection (GPF) [49] and support vector machines (SVM) to classify estimates of eye centers. Bai *et al*. [3] exploit an enhanced version of Reisfeld's generalized symmetry transform [39] for the task of eye location. Wang *et al*. [47], [48] use statistically learned non-parametric discriminant features combined into weak classifiers, using the AdaBoost algorithm. Hamouz *et al*. [24] search for ten features using Gabor filters, use features triplets to generate face hypothesis, register them for affine transformations, and finally verify the remaining configurations using two SVM classifiers. Campadelli *et al*. [7] employ an eye detector to validate the presence of a face and to initialize an eye locator, which in turn refines the position of the eye using SVM on optimally selected Haar wavelet coefficients. Duffner [16] makes use of convolutional neural networks. The method by Niu *et al*. [36] uses a iteratively bootstrapped boosted cascade of classifiers. Kim *et al*. [30] discuss a multi-scale approach to localize eyes based on Gabor vectors. Wang *et al*. [46] treat faces as a 3D landscape, and they use the geometric properties of this terrain to extract potential eye regions. These candidates are then paired and classified using a Bhattacharyya kernel based SVM. Huang and Wechsler [26] also treat the face image as a landscape, where final state automata are genetically evolved to walk the landscape and derive a saliency map for the best plausible location of the eyes. These salient regions are then classified as eyes by using genetically evolved decision trees. Reale *et al*. [38] map the 2D eye texture to a 3D eye ball, then fit a circle to the iris to find the optimal eye ball rotation. Asadifard and Shanbezadeh [1] filter the eye image to find pixel values which are likely to belong to the pupil in an adaptive manner. Timm and Barth [41] use the gradient field to find the most probable eye center. Finally, Kroon *et al*. [32] apply a Fisher Linear Discriminant to filter the face image and select the highest responses as eye center.

A summary of the characteristics of the discussed literature is presented in Table 1.

As indicated by the last lines of Table 1, this paper will describe a feature based eye center locator which can quickly, accurately, and robustly locate eye centers in low resolution images and videos (*i.e*. coming from a simple web cam). Further, this paper will also show how the method is easily extended into a hybrid approach. Hence, we made the following contributions:

- A novel eye location approach is proposed, which is based on the observation that eyes are characterized by radially symmetric brightness patterns. Contrary to other approaches using symmetry to accomplish the same task [3], our method makes use of isophotes (Section 2) to infer the center of (semi)circular patterns and gain invariance to in-plane rotation and linear lighting changes.
- A novel center voting mechanism (Section 3) based on gradient slope is introduced in the isophote framework to increase and weight important votes to reinforce the center estimates.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.



(a) (b) (c)

Fig. 1. The original image (a), its isophote curvature at the edges (b), and the 3D plot of the latter (c).

(a) (b) (c)

Fig. 2. A detail showing the direction of the gradient under the image's edges (a), the displacement vectors pointing to the isophote centers (b), and the centermap (c).

- The integration of our method in a scale space framework to find the most stable results.

In this paper we study the accuracy and the robustness of the proposed approach to lighting, occlusion, pose and scale changes, and compare the obtained results with the state of the art systems for eye location in standard (*i.e.* visible wavelength), low resolution imagery (Section 4).

2 Isophotes Curvature Estimation

The iris and pupil are very prominent circular features which are characterized by an approximately constant intensity along the limbus (the junction between the sclera and the iris), and the iris and the pupil. We can therefore represent these features using isophotes, which are curves connecting points of equal intensity (one could think of isophotes as contour lines obtained by slicing the intensity landscape with horizontal planes). Since isophotes do not intersect each other, an image can be fully described by its isophotes. Furthermore, the shape of the isophotes is independent to rotation and linear lighting changes [33]. Due to these properties, isophotes have been successfully used as features in object detection and image segmentation [18], [29], [33].

To better illustrate the isophote framework, the notion of intrinsic geometry is introduced, *i.e.* geometry with a locally defined coordinate system. In every point of the image, a local coordinate frame is fixed in such a way that it points in the direction of the maximal change of the intensity, which corresponds to the direction of the gradient. This reference frame $\{v, w\}$ is also referred to as the *gauge coordinates*. Its frame vectors $\hat{w}$ and $\hat{v}$ are defined as:

$$\hat{w} = \frac{\{L_x, L_y\}}{\sqrt{L_x^2 + L_y^2}}; \hat{v} = \perp \hat{w}; \tag{1}$$

where L_x and L_y are the first-order derivatives[1] of the luminance function $L(x, y)$ in the x and y dimension, respectively. In this setting, a derivative in the w direction is the gradient itself, and the derivative in the v direction (perpendicular to the gradient) is 0 (no intensity change along the isophote).

In this coordinate system, an isophote is defined as $L(v, w(v)) = \text{constant}$ and its curvature is defined as the change w'' of the tangent vector w'. By implicit differentiation with respect to v of the isophote definition, we obtain:

$$L_v + L_w w' = 0; \qquad w' = -\frac{L_v}{L_w}. \tag{2}$$

Since $L_v = 0$ from the gauge condition, then $w' = 0$. Differentiating again with respect to v, yields

$$L_{vv} + 2L_{vw}w' + L_{ww}w'^2 + L_w w'' = 0. \tag{3}$$

Solving for $\kappa = w''$ (the isophote curvature) and recalling that $w' = 0$, the isophote curvature is obtained as

$$\kappa = -\frac{L_{vv}}{L_w}. \tag{4}$$

In Cartesian coordinates, this becomes [14], [44], [25]

$$\kappa = -\frac{L_{vv}}{L_w} = -\frac{L_y^2 L_{xx} - 2L_x L_{xy} L_y + L_x^2 L_{yy}}{(L_x^2 + L_y^2)^{3/2}}. \tag{5}$$

To better illustrate the effect of the theory on an image, a simplistic eye model is used, shown in Figure 1(a). The isophote curvature of the eye model is shown in Figure 1(b). For presentation purposes, the shown curvature belongs to the isophote under the edges found in the image using a Canny operator. The crown-like shape of the values in the 3D representation (Figure 1(c)) is generated by the aliasing effects due to image discretization. By scaling[2] the original image this effect is reduced, but at higher scales the isophotes curvature might degenerate with the inherent effect of losing important structures in the image.

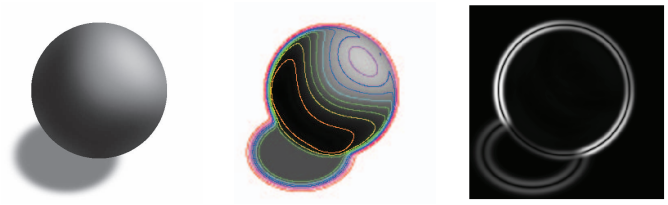
3 Isophote Centers

For every pixel, we are interested in retrieving the center of the circle which fits the local curvature of the isophote. Since the curvature is the reciprocal of the radius, Eq. (5) is reversed to obtain the radius of this circle. The obtained radius magnitude is meaningless if it is not combined with orientation and direction. The orientation can be estimated from the gradient, but its direction will always point towards the highest change in the luminance (Figure 2(a)). However, the sign of the isophote curvature depends on the intensity of the outer side of the curve (for a brighter outer side the sign is positive). Thus, by multiplying the gradient with the inverse

1. In our implementation, we use the fast anisotropic Gauss filtering method proposed in [20] to compute image derivatives. The used sigma parameter is equal in both direction (isotropic), with a rotation angle of 0°

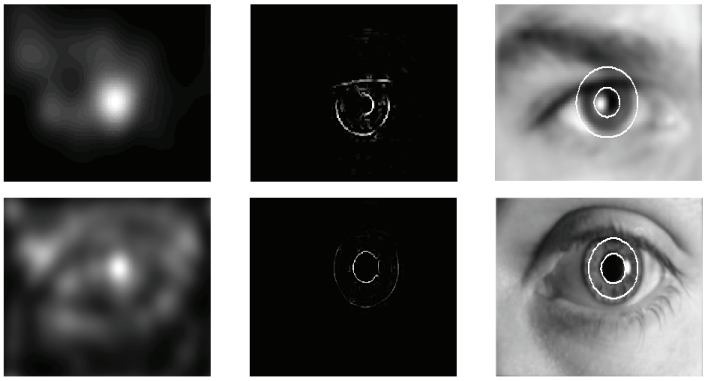
2. Scale in this context represents the standard deviation of the Gaussian kernel or its derivatives with which the image is convolved. See [14], [31] for more details.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.



(a) (b) (c)

Fig. 3. A sphere illuminated from above and casting a shadow (a), a sample of the isophotes superimposed to the image (b), the curvedness value of the same image (c).



(a) (b) (c)

Fig. 4. The obtained centermap (a), the edges that contributed to the vote of the MIC (b), the average of the two biggest clusters of radiuses which voted for the found MIC (c).

of the isophote curvature, the sign of the isophote curvature helps in disambiguating the direction to the center. Since the unit gradient can be written as $\frac{\{L_x,L_y\}}{L_w}$, we have

$$\begin{aligned}\{D_x,D_y\} &= \frac{\{L_x,L_y\}}{L_w}\left(-\frac{L_w}{L_{vv}}\right) = -\frac{\{L_x,L_y\}}{L_{vv}} \\ &= -\frac{\{L_x,L_y\}(L_x^2+L_y^2)}{L_y^2L_{xx}-2L_xL_{xy}L_y+L_x^2L_{yy}}. \end{aligned} \quad (6)$$

where $\{D_x,D_y\}$ are the displacement vectors to the estimated position of the centers, which can be mapped into an accumulator, hereinafter *"centermap"*. Note that sometimes the isophote curvature could assume extremely small or big values. This indicates that we are dealing with a "straight line" or a "single dot" isophote. Since the estimated radius to the isophote center would be too high to fall into the centermap or too little to move away from the originating pixel, the calculation of the displacement vectors in these extreme cases can simply be avoided. The set of vectors pointing to the estimated centers are shown in Figure 2(b). When compared to Figure 2(a), it is possible to note that the vectors are now all correctly directed towards the center of the circular structures. Figure 2(c) represents the cumulative vote of the vectors for their center estimate (*i.e.* the accumulator). Since every vector gives a rough estimate of the center, the accumulator is convolved with a Gaussian kernel so that each cluster of votes will form a single center estimate. The contribution of each vector is weighted according to a relevance mechanism, discussed in the following section.

3.1 Center Voting

So far an edge-based approach and a simplistic eye model were used to ease the explanations. Instead of using the peaks of the gradient landscape (*i.e.* edges), we propose to use the slope information around them, as they contain much more information.

In the simplistic eye model in Figure 1(a) there are only three isophotes: one describing the pupil, one describing the iris and one describing the boundary of the sclera. By convolving the eye model with a Gaussian kernel, it can be observed that the number of isophotes increases around the edges as the steepness of the edge decreases, and that each of these new isophotes is similar to the original isophotes (besides some creations and annihilations), so they can be used to generate additional evidence to vote for a correct center. The main idea is that by collecting and averaging local evidence of curvature, the discretization problems in a digital image could be lessened and an invariant and accurate eye center estimation could be achieved.

Contrary to the shown example, in real world environments there are no guarantees that the boundaries of an object are of the same intensity, *i.e.* that there is a sole isophote under the object's edges. In this case, allowing every single isophote to vote for a center will produce meaningless results since, due to highlights and shadows, the shape of the isophotes significantly differs from the shape of the object (Figure 3(a)(b)). In order to cope with this drawback, only the parts of the isophotes which are meaningful for our purposes should be considered.

To this end, an image operator that indicates how much a region deviates from flatness is needed. This operator is the curvedness [31], defined as

$$\text{curvedness} = \sqrt{L_{xx}^2+2L_{xy}^2+L_{yy}^2}. \quad (7)$$

The curvedness can be considered as a rotational invariant gradient operator, which measures the degree of steepness of the gradient. Therefore, it yields low response on flat surfaces and edges, whereas it yields high response around the edges (Figure 3(c)). Since isophotes are slices of the intensity landscape, there is a direct relation between the value of the curvedness and the density of isophotes. Therefore, denser isophotes are likely to belong to the same feature (*i.e.* edge) and thus locally agree on the same center. A comparison between Figures 3(b) and 3(c) shows this relation between the curvedness and the image isophotes. It is clear that the curvedness is higher where the isophotes are denser. Therefore, by only considering the isophotes where the curvedness is maximal, they will likely follow an object boundary. The advantage of the proposed approach over a pure edge based method is that, by using the curvedness value as the weighting scheme for the importance of the vote, every pixel in the image

may be used to contribute to a final decision. By summing the votes, we obtain high responses around the center of isocentric isophotes patterns. We call these high responses *"isocenters"*, or ICs. The maximum isocenter (MIC) in the centermap will be used as the most probable estimate for the soughtafter location of the center of the eye.

3.2 Eye Center Location

Recalling that the sign of the isophote curvature depends on the intensity of the outer side of the curve, it can be observed that a negative sign indicates a change in the direction of the gradient (*i.e*. from brighter to darker areas). Therefore, it is possible to discriminate between dark and bright centers by analyzing the sign of the curvature. Regarding the specific task of pupil and iris location, it can be assumed that the sclera is brighter than the iris and the pupil, therefore the votes which move from darker to brighter areas (*i.e*. in which the curvature agrees with the direction of the gradient), can be simply ignored in the computation of the isocenters. This allows the method to cope with situations in which strong highlights are present (*e.g*. when using an infrared illuminator or in the eye images in Figure 4), as long as the circular pattern of the eye is not heavily disrupted by them. Once the MIC is found, it is possible to retrieve a distribution of the most relevant radii (*i.e*. the pupil and the iris) by clustering together the distance to the pixels which voted for it. Figure 4 shows the results of the procedure applied on two high resolution images of eyes. Note from Figure 4(b) that the vote contribution coming from highlights are not considered in the computation of the MIC.

3.3 Eye Center Location: Scale and Scale Space

Although the proposed approach is invariant to rotation and linear illumination changes, it still suffers from changes in scale. While in the previous work [43] the scale problem was solved by exhaustively searching for the scale value that obtained the best overall results, here we want to gain scale independence in order to avoid adjustments to the parameters for different situations. Firstly, since the sampled eye region depends on the scale of the detected face and on the camera resolution, to improve scale independency each eye region is scaled to a reference window. While this technique is expected to slightly decrease the accuracy with respect to the previously proposed approach (due to interpolation artifacts), once the correct scale values are found for the chosen reference window, the algorithm can be applied at different scales without requiring an exhaustive parameter search.

Furthermore, to increase robustness and accuracy, a scale space framework is used to select the isocenters that are stable across multiple scales. The algorithm is applied to an input image at different scales and the outcome is analyzed for stable results. To this end, a Gaussian pyramid is constructed from the original grayscale image. The image is convolved with different Gaussians so that they are separated by a constant factor in scale space. In order to save computation, the image is downsampled into octaves. In each octave the isocenters are calculated at different intervals: for each of the image in the pyramid, the proposed method is applied by using the appropriate σ as a parameter for image derivatives. In our experiments (Section 4), we used three octaves and three intervals for each octave (as in [34]). This procedure results in a isocenters pyramid (Figure 5). The responses in each octave are combined linearly, then scaled to the original reference size to obtain a scale space stack. Every element of the scale space stack is considered equally important therefore they are linearly summed into a single centermap. The highest peaks in the resulting centermap will represent the most scale invariant isocenters.

3.4 Eye Center Location: Mean Shift and Machine Learning

Although the MIC should represent the most probable location for the eye center, certain lighting conditions and occlusions from the eyelids are expected to result in a wrong MIC. In order to avoid obtaining other isocenters as eye center estimates, two additional enhancements to the basic approach presented in the previous section are proposed, the first using mean shift for density estimation and the second using machine learning for classification.

Mean shift (MS) usually operates on back-projected images in which probabilities are assigned to pixels based on the color probability distribution of a target, weighted by a spatial kernel over pixel locations. It then finds the local maximum of this distribution by gradient ascent [10]. Here, under the assumption that the most relevant isocenter should have higher density of votes, the mean shift procedure is directly applied to the centermap as if it was a distribution. Since wrong MICs are not so distant from the correct one (*e.g*. on an eye corner), the mean shift search window is initialized centered on the found MIC, with dimensions equal to half the detected eye region's height and width. The algorithm then iterates to climb the centermap and converge to a region with maximal density of center votes. The isocenter closest to the center of the converged search window is then selected as the new eye center estimate.

Machine Learning: instead of considering the strongest isocenter as eye center estimate, the aim is to consider the n most relevant ones and to discriminate between them using any classification framework. In this way, the task of the classifier is simplified as it only has to deal with a two class problem (eye center or not) and to discriminate between a couple of features (centered on the n most relevant isocenters). Note that the final performance of the system will always be bounded by the quality of the candidate locations (more on this in Section 4.3). For our experimentation, two different input features are used, centered on the candidate isocenters: 1) the pixel intensity sampled from a fixed window (dimensions depending on the detected face boundary) scaled to a 256 dimensional feature vector and 2) a SIFT [34] based descriptor, which differs from the SIFT as it does not search for scale invariant features, since the location of the feature is already known. Removing invariances from SIFT in an application-specific way has been shown to improve accuracy in [40].

The reasoning behind the choice of these two specific features is that 1) intensity is a rich source of information,

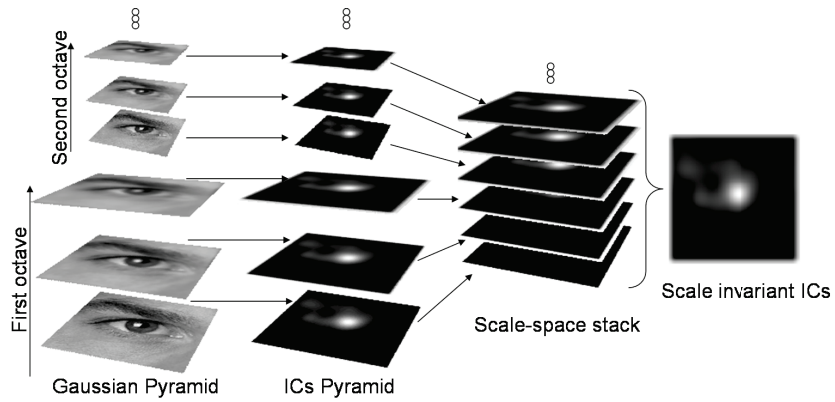


Fig. 5. The scale space framework applied to eye location: the grayscale image is downscaled to different octaves, each octave is divided into intervals. For each interval, the centermap is computed and upscaled to a reference size to obtain a scale space stack. The combination of the obtained results gives the scale invariant isocenters.

Method	Pixels	Sift
Fisher Discriminant	14.05%	10.82%
Nearest Mean	30.75%	14.02%
Scaled nMean	30.38%	13.54%
Parzen	7.36%	6.92%
Neural Network	25.00%	29.38%
kNN	7.78%	**6.10%**

TABLE 2
Mean errors obtained by some of the tested classification methods on the raw pixels and the SIFT-like descriptor.

and should naturally be included as baseline and 2) SIFT features have been shown to yield good generalization due to the reduced feature space and robustness. Both descriptors are computed on the original image, centered on the location of each of the candidate isocenters. Afterwards, the obtained descriptors are scaled to a reference size.

The following classification frameworks were selected to be representative of different classification approaches which are sensible to the selected features and method [17]: A one-against-all linear Fisher discriminant; A nearest mean and a scaled nearest mean classifier (in which the features are scaled to fit a normal distribution); A Parzen density estimator with feature normalization for each class based on variance; An automatically trained feed-forward neural network classifier with a single hidden layer; A kNN classifier, where k is optimized with respect to the leave-one-out error obtained from the database.

For the sake of completeness, the obtained classification results are shown in Table 2. We have used 10-fold cross-validation in each experiment on the BioID database (described in the next section), where both training and validation folds are actually selected from the original training set. The test set, on which we report our overall results, is not seen during cross-validation. Given the simplicity of the problem, it is not surprising that the kNN classifier with the more robust SIFT-based descriptor is able to achieve the best results (Table 2). This is because the features are extracted around a point suggested by our method, hence it is quite likely that the training and testing feature vectors will not be exactly aligned (*e.g.* it not always centered on the eye center or the eye corner). Hence, the robustness of the feature descriptor to minor perturbations from the target location plays an important role, and SIFT provides for this by allowing overlaps in shifted vectors to result in meaningful similarity scores. In view of the high accuracies, computational cost is also a major guiding factor, hence the combination of the SIFT based feature and the kNN classification framework is used in the evaluation as an example of a hybrid variant of our method.

4 Evaluation

So far, high resolution images of eyes have been used as examples. In this section, the proposed method is tested on low resolutions eye images, *e.g.* coming from face images captured by a web cam. Additionally, the method is tested for robustness in changes in pose, illumination, scale and occlusion.

4.1 Procedure and Measures

In order to obtain low resolution eye images from face images in the used test sets, the face position of each subject is estimated by using the boosted cascade face detector proposed by Viola and Jones [45][3]. The rough positions of the left and right eye regions are then estimated using anthropometric relations[4]. The proposed procedure is then applied to the cropped eye regions in order to accurately locate the center of the eye.

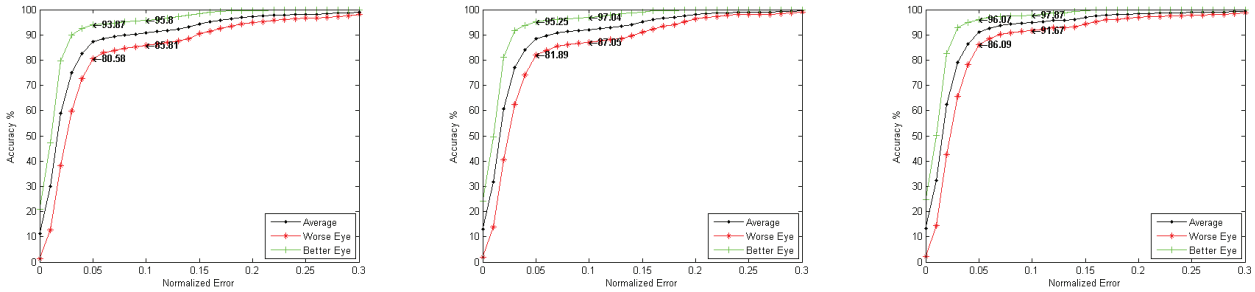
The *normalized error*, indicating the error obtained by the worse eye estimation, is adopted as the accuracy measure for the found eye locations. This measure was proposed by Jesorsky *et al.* [27] and is defined as:

$$e = \frac{\max(d_{\text{left}}, d_{\text{right}})}{w}, \tag{8}$$

3. The OpenCV implementation with default parameters is used in our experiments, discarding false negatives from the test set

4. We empirically found that, in the used datasets, eye centers are always contained within two regions starting from 20%x30% (left eye) and 60%x30% (right eye) of the detected face region, with dimensions of 25%x20% of the latter.

Fig. 6. Sample of success (first row) and failures (second row) on the BioID face database; a white dot represents the estimated center.



(a) MIC (b) MIC+MS (c) MICs+ SIFT kNN

Fig. 7. Accuracy vs. minimum (better eye) and maximum (worse eye) normalized error obtained by the proposed methods on the BioID database.

where d_{left} and d_{right} is the Euclidean distance between the found left and right eye centers and the ones in the ground truth, and w is the Euclidean distance between the eyes in the ground truth. In this measure, $e \leq 0.25$ (a quarter of the interocular distance) roughly corresponds to the distance between the eye center and the eye corners, $e \leq 0.1$ corresponds to the range of the iris, and $e \leq 0.05$ corresponds the range of the pupil. To give upper and lower bounds to the accuracy, in our graphs (Figures 7 and 9) the *minimum normalized error* (obtained by considering the better eye estimation only) and an average between the better and worse estimation are also shown. These values are also needed in order to compare our results with other published works which make use of the normalized error measure in a non standard way.

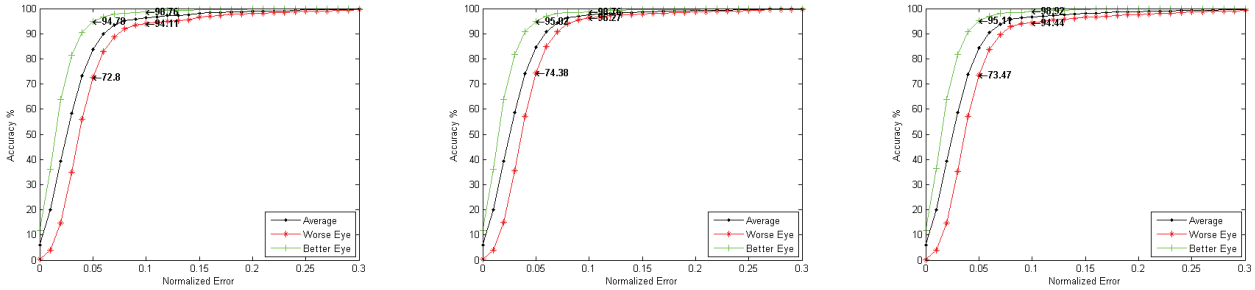
4.2 Results

The BioID [5] and the color FERET [37] databases are used for testing. The BioID database consists of 1521 grayscale images of 23 different subjects and has been taken in different locations and at different times of the day (*i.e.* uncontrolled illumination). Besides changes in illumination, the positions of the subjects change both in scale and pose. Furthermore, in several samples of the database the subjects are wearing glasses. In some instances the eyes are closed, turned away from the camera, or completely hidden by strong highlights on the glasses. Due to these conditions, the BioID database is considered a difficult and realistic database. The size of each image is 384x288 pixels. A ground truth of the left and right eye centers is provided with the database.

The color FERET database contains a total of 11338 facial images collected by photographing 994 subjects at various angles, over the course of 15 sessions between 1993 and 1996. The images in the color FERET Database are 512 by 768 pixels. In our case we are only interested in the accuracy of the eye location in frontal images, therefore only the frontal face (fa) and alternate frontal face (fb) partitions of the database are considered. Figure 6 and Figure 8 show the qualitative results obtained on different subjects of the BioID and the color FERET databases, respectively. We observe that the method successfully deals with slight changes in pose, scale, and presence of glasses. By analyzing the failures (second row) it can be observed that the system is prone to errors when the circular eye pattern is altered by the presence of closed eyelids or strong highlights on the glasses. When these cases occur, the iris and pupil do not contribute enough to the center voting, so the eyebrows or the eye corners assume a position of maximum relevance.

The graphs in Figure 7(a) and Figure 9(a) quantitatively show the accuracy of our method for different e. While it is clear that most of the results are nearly optimal, there is a saddle on the normalized error around the value of 0.15. This clustering of errors proves that few errors occur between the real eye centers and the eye corners/eyebrows. The improvement obtained by using the mean shift procedure

Fig. 8. Sample of success (first row) and failures (second row) on the color FERET face database; a white dot represents the estimated center, while a red dot represents the human annotation.



(a) MIC (b) MIC+MS (c) MICs+ SIFT kNN

Fig. 9. Accuracy vs. minimum (better eye) and maximum (worse eye) normalized error obtained by the proposed methods on the color FERET database.

for maximum density can be seen by comparing the graphs in Figures 7(a) and (b). Without any additional constraint, the results improved with $\approx 2.5\%$ over the basic approach. The graphs in Figure 7 (c) and 9 (c) show the accuracy obtained by using the kNN classifier to discriminate between the top MICs, which in case of the BioID database achieved better results than both the basic and the mean shift approaches, while the results on the color FERET database show a slight drop in accuracy, which becomes comparable to the basic approach. This can be explained by the fact that by using classification the successful outcome of the system will inherently depend on the conditions it was trained, together with the fact that the annotation in the color FERET database is sometimes unreliable. In fact, it can be seen from Figure 8 that the human annotation (indicated by a red dot) is sometimes less accurate than the estimated eye center (indicated by a white dot). This negatively affects the accuracy for accurate eye center location and its effect can be seen by comparing the graphs in Figure 9 to the ones in Figure 7: the differences between the results at $e \leq 0.05$ and the ones at $e \leq 0.1$ are significantly higher than the ones found on the BioID database.

4.3 Comparison with the State of the Art

Our results are compared with state of the art methods in the literature which use the same databases and the same accuracy measure. While many recent results are available on the BioID database, results on the color FERET database are often evaluated on custom subsets and with different measures, therefore not directly comparable. This is the case of Kim *et al.* [30] which only use 488 images of the "fa" subset (frontal face, neutral expression) and of Duffner [16] which, instead of using the maximum error measure as in this paper, evaluates the normalized error on both eyes instead of the worse one only. This is equivalent to the "Average" curves in Figure 9 where the best variant (MIC+MS) obtains an accuracy of 85.10% for $e \leq 0.05$ versus Duffner's 79.00%. Tables 3 and 4 show the comparison between our methods and the state of the art methods mentioned in Section 1 for several allowed normalized errors. Where inexplicitly reported by the authors, the results are estimated from their normalized error graphs, safely rounded up to the next unit. It can be seen that, for an allowed normalized error smaller than 0.25, we achieved accuracy comparable to the best methods. For iris location ($e \leq 0.1$), our method shows less accuracy with respect to the some of the other methods. This can be justified by the fact

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

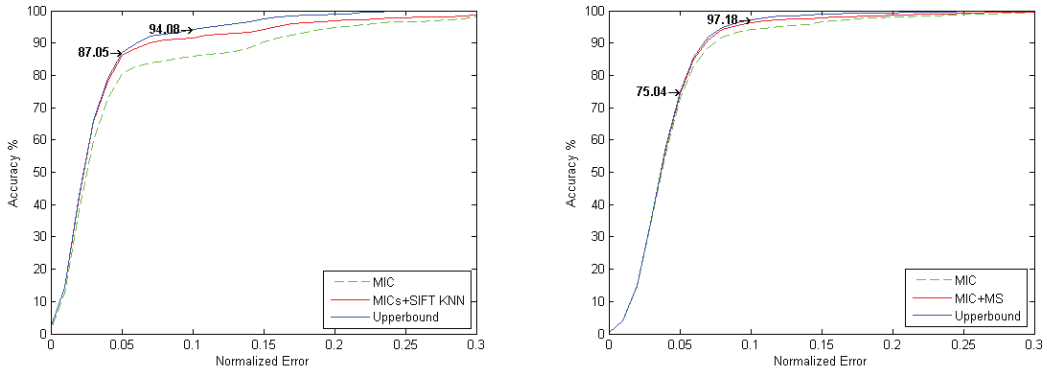


Fig. 10. A summary of the better and worse results obtained on the BioID and on the color FERET databases in comparison with the respective upper bound curves.

Method	Accuracy ($e \leq 0.05$)	Accuracy ($e \leq 0.10$)	Accuracy ($e \leq 0.25$)
MIC	80.58%	85.81%	96.56%
MIC+MS	81.89%	87.05%	**98.00%**
MICs+SIFT	**86.09%**	**91.67%**	97.87%
Asteriadis [2]	74.00%*	81.70%	97.40%
Jesorsky [27]	40.00%	79.00%	91.80%
Cristinacce [13]	56.00%*	96.00%	98.00%
Türkan [42]	19.00%*	73.68%	**99.46%**
Bai [3]	37.00%*	64.00%	96.00%
Campadelli [7]	62.00%	85.20%	96.10%
Hamouz [24]	59.00%	77.00%	93.00%
Kim [30]	n/a	**96.40%**	98.80%
Niu [36]	**75.00%***	93.00%	98.00%*
Asadifard [1]	47.00%	86.00%	96.00%
Timm [41]	82.50%	93.40%	98.00%
Kroon [32]	65.00%*	87.00%	98.80%*

TABLE 3
Comparison of accuracy vs. normalized error in the BioID database. *= the value estimated from author's graphs.

Method	Accuracy ($e \leq 0.05$)	Accuracy ($e \leq 0.10$)	Accuracy ($e \leq 0.25$)
MIC	72.80%	94.11%	98.21%
MIC+MS	**74.38%**	**96.27%**	**99.17%**
MICs+SIFT	73.47%	94.44%	98.34%
Campadelli [7]	67.70%	89.50%	96.40%
Duffner [16]	79.00%*	97.00%*	99.00%*
Kim [30]	91.80% ($e \leq 0.07$)		

TABLE 4
Comparison of accuracy vs. normalized error in the color FERET database.*= uses average normalized error.

that the other methods exploit other facial features to estimate and adjust the position of the eyes (*i.e.* the eye center is in between the eye corners) which works extremely well to find a point in the middle of two eye corners, but often does not have enough information to locate the exact position eye center in between them. However, our approach excels for accurate eye center location ($e \leq 0.05$), even by using the basic approach.

To measure the maximum accuracy achievable by our method, we computed the normalized error obtained by selecting the isocenter closest to the ground truth. The graphs in Figure 10 show the comparison between the better and worse performing variants of the proposed method and an additional curve which represents the found upper bound on the BioID and color FERET databases. It is possible to see that the proposed extensions helped in increasing the bending point of the curve, while the rest of the curve is similar in all the cases. This means that the extensions reduced the number of times an eye corner or an eyebrow is detected as the MIC, moving the results closer to the upper bound. Note that the SIFT extension almost follows the upper bound for $e \leq 0.05$.

4.4 Robustness to Illumination and Pose Changes

To systematically evaluate the robustness of the proposed eye locator to lighting and pose changes, two subsets of the Yale Face Database B [19] are used. The full database contains 5760 grayscale images of 10 subjects each seen under 576 viewing conditions (9 poses x 64 illuminations). The size of each image is 640x480 pixels. To independently evaluate the robustness to illumination and pose, the system is tested on frontal faces under changing illumination (10 subjects x 64 illuminations) and on changing pose under ambient illumination (10 subjects x 9 poses).

The first two rows of Figure 11 show a qualitative sample of the results obtained for a subject in the illumination subset. By analyzing the results, we note that the system is able to deal with light source directions varying from $\pm 35°$ azimuth and from $\pm 40°$ elevation with respect to the camera axis. The results obtained under these conditions are shown in Table 5. When compared to the previously published results in [43], the improvement in accuracy obtained by the scale space framework is about 2%, especially for the MS extension. For higher angles, the method is often successful for the less illuminated eye and sporadically for the most illuminated one: if the eye is uniformly illuminated, its center is correctly located, even for low intensity images; if, on the other hand, the illumination influences only parts of the eye, the shape of the isophotes is influenced by shadows, resulting in an unreliable MIC.

The last row in Figure 11 shows the results of the eye locator applied to a subject the pose subset of the Yale Face Database B. The quantitative evaluation on this dataset shows the robustness of the proposed approach to pose changes: due to the higher resolution and the absence of occlusions and glasses, all the variants achieved an accuracy of 100.00% for $e \leq 0.05$. The first errors are actually found by considering $e \leq 0.04$ for the basic method (MIC), where the system achieves an accuracy of 95.45%.

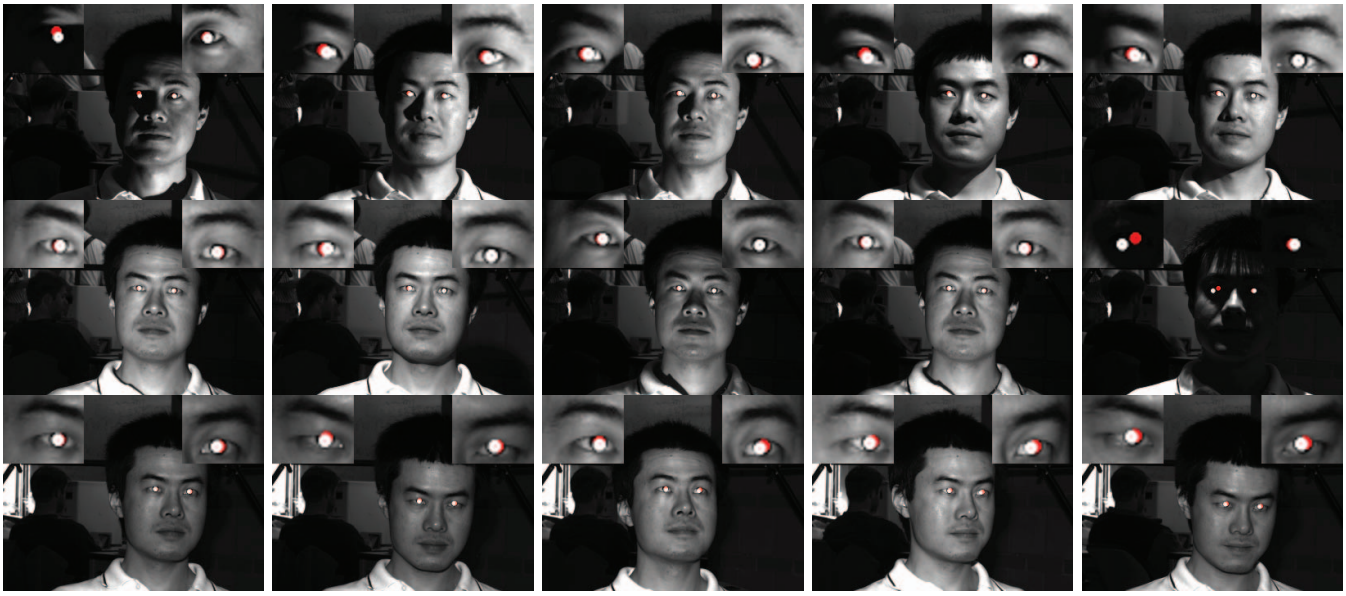


Fig. 11. Effect of changes in illumination and pose (last row) on a subject of the Yale Face Database B.

Fig. 12. Effect of changes in illumination (horizontally) and pose (vertically) on a subject of the Multi-PIE database.

Method	Accuracy ($e \leq 0.05$)	Accuracy ($e \leq 0.10$)	Accuracy ($e \leq 0.25$)
MIC	77.68%	85.32%	95.72%
MIC+MS	79.82%	88.07%	96.64%
MICs+SIFT	80.12%	86.85%	96.73%

TABLE 5
Accuracy vs. normalized error for illumination changes on the Yale Face Database B.

To systematically evaluate the combined effect of lighting and pose changes, the CMU Muliti-PIE database [23] is used. The database contains images of 337 subjects, captured under 15 view points and 19 illumination conditions in four recording sessions for a total of more than 750,000 images. The database shows very challenging conditions for the proposed method, as many subjects have closed eyes due to the natural reaction to flashes, or the irises are occluded due to very strong highlights on the glasses, generated by the flashes as well.

As no eye center annotation is provided with the database, we manually annotated the eye centers and the face position of all the subjects in the first session (249), in 5 different poses (the ones in which both eyes are visible), under all the different illumination conditions present in the database. This annotation is made publicly available on the author's website. Figure 12 shows a qualitative sample of the database, together with the annotation and obtained result. Table 6 and the interpolated 3D plot in Figure 13 quantitatively show the result of this experiment for $e \leq 0.05$, using the MIC+MS

	Illumination												
Pose	$-90°$	$-75°$	$-60°$	$-45°$	$-30°$	$-15°$	$0°$	$+15°$	$+30°$	$+45°$	$+60°$	$+75°$	$+90°$
$-30°$	70.28%	78.71%	83.13%	82.33%	84.74%	89.56%	91.97%	94.38%	95.58%	89.56%	74.70%	58.23%	51.00%
$-15°$	66.67%	78.31%	82.73%	87.95%	88.76%	91.57%	96.39%	97.19%	97.99%	94.78%	81.12%	57.43%	52.61%
$0°$	73.09%	78.31%	83.94%	89.96%	89.16%	95.58%	93.17%	98.80%	98.80%	97.59%	89.56%	71.89%	61.45%
$+15°$	62.25%	71.89%	78.71%	91.16%	92.37%	97.19%	95.18%	96.79%	96.79%	95.18%	88.76%	77.51%	64.66%
$+30°$	36.55%	51.41%	59.84%	79.12%	84.34%	91.16%	89.96%	87.95%	90.36%	85.54%	81.53%	73.09%	68.27%

TABLE 6
Combined effect of changes in head pose and illumination in the Multi-PIE database for $e \leq 0.05$, using MIC+MS.

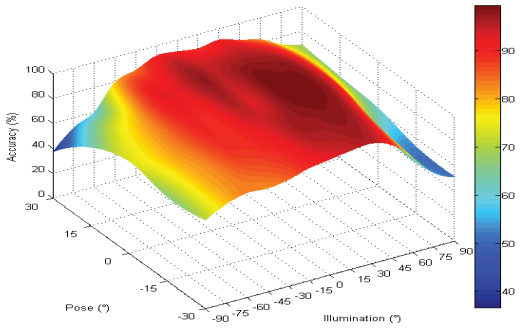


Fig. 13. A interpolated 3D representation of the data in Table 6.

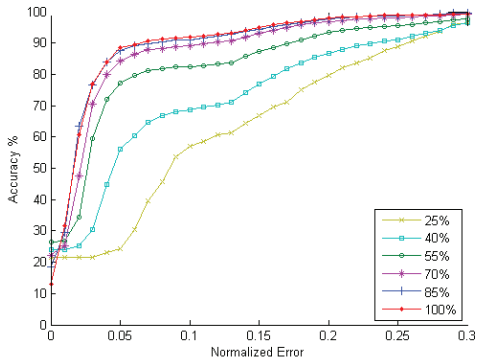


Fig. 15. The effect of scaling down of the images on the BioID database, at different percentages of the original size.

variant. As with the YALE Face Database B, this variant obtained better results with respect to the MICs+SIFT variant due to the variance present in the training data, which makes it difficult for the classifier to find a clear decision boundary to discriminate eye centers from the rest of the features.

By analyzing the results, it is possible to derive insights about the accuracy, the success and failures of the proposed method: Although the frontal face with frontal illumination is expected to achieve the best accuracy, the fact that the flash directly reflects on subjects wearing glasses contributes to a drop in accuracy in that specific setting. However, if the illumination is shifted by just $15°$, the system is able to achieve an accuracy of 98.80%, which is the best result obtained in this experiment. Furthermore, it is possible to note that the accuracy is higher when the face is turned towards the light. This is because the shape of the irises in these situations will not be affected by shadows. This behavior is very clear from the 3D plot in Figure 13.

4.5 Robustness to Scale Changes

The system uses only two parameters: the "scale" of the kernel (σ_{total}) with which the image derivatives are computed and the "scale" of the Gaussian kernel with which the centermap is convolved (*i.e.* how much near votes affect each other). Figure 14(a) shows the changes in accuracy for different values of σ_{total}. It can be seen that, by changing this parameter, the curves shift vertically, therefore the value that results in the highest curve should be selected as the best σ_{total} (in this case, 3). This is not the case with the graph in Figure 14(b) which shows the effect of changing the blurring parameter of the centermap (*i.e.* how near votes affect each other). In this case, the accuracy remains basically unchanged for accurate results ($e \leq 0.04$), while selecting a proper kernel size (*e.g.* 16) improves the bending point of the curve (*i.e.* the errors between the eye centers and eye corners). In order to study the effect of changing the scale now that the best parameters are known, the test images are downscaled to fixed ratio values: 25%, 40%, 55%, 70%, 85% and 100% of the original image size. The eyes are then cropped and upscaled to a reference window size (*e.g.* 60x50 pixels) where the best value of the size of the Gaussian kernel for the image derivatives is experimentally known. The scale space isocenters pyramid is then computed with a value of σ^2 at interval i calculated by

$$\sigma^2_{\text{total}} = \sigma^2_i + \sigma^2_{i-1}, \tag{9}$$

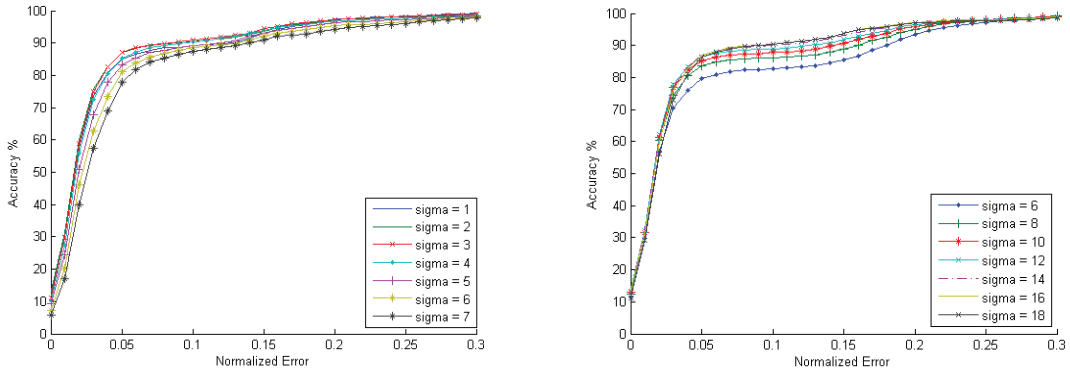
therefore

$$\sigma_i = \sqrt{\sigma^2_{\text{total}} - \sigma^2_{i-1}}. \tag{10}$$

The result of this experiment is shown in Figure 15. Note that downscaling from 100% to 85% and to 70% does not significantly affect the results, while the rest of the results are still acceptable considering the downsampling artifacts and the size of the images.

4.6 Robustness to Occlusions

Since the proposed method is based on the assumption that the eye pattern is circular and that is visible, it is important to evaluate the robustness of the approach to partial occlusion which might result from eye blinking, facial expressions and extreme eye positions. Since many subjects in the BioID database display closed or semi-closed eyes, the obtained overall accuracy can already give an indication that the proposed approach is able to handle eye occlusion. To validate the robustness to occlusion of the proposed method, a simple



(a) (b)

Fig. 14. The effect of changing the parameters on the average normalized error using the BioID database. Changing the size of (a) Gaussian kernel for image derivatives (b) Gaussian kernel for the centermap.

experiment was performed on 10 subjects. The subjects were requested to gaze at the camera and slowly close their eyes. The system recorded the first image in which the eye center estimation would move by more than 5% of the interocular distance from their initial position. A sample of the results is shown in Figure 16, where it is clear that the system is able to handle situations in which the iris is almost completely occluded.

To give a better overview of the behavior of our method to progressive occlusions, we designed a procedural experiment that simulates eye occlusions by a shifting rectangle. The color of the rectangle was sampled from the average color of eyelids in the database. Note that, since the rectangle's edges are straight (null curvature), the votes generated by the rectangle are automatically discarded by the method and will not affect the center detection. In order to analyze every percentage of occlusion, a subset of the subjects displaying completely open eyes where selected from the BioID database. In our experiment, we define a 0% eye occlusion when the lower side of the occluding rectangle touches the uppermost point the iris, a 50% occlusion when it passes through the middle of the pupil, and a 100% occlusion when is tangent to the lowest point in of the iris. The graph in Figure 18 shows that the proposed method can successfully detect eye centers even if they are occluded by more than 60%. In fact, up to 50% occlusion, the method degrades in accuracy only by less than 10% for accurate eye center location ($e \leq 0.05$). An insight that arises from this experiment is that at 100% occlusion the system is sometimes able to locate the eye center. This is because the closed eye region is generally darker than the features around the eye, and therefore it can still generate votes which fall into the pupil area. An example of the occlusion procedure in this experiment is shown in Figure 17. Note that, since the centermap is always normalized, it does not seem to change significantly. However, it is possible to see that the found MIC moves down and that the right eye corner gains more votes as the circular iris pattern disappears.

To systematically evaluate the robustness of the proposed approach to eye rotation, an additional experiment in which

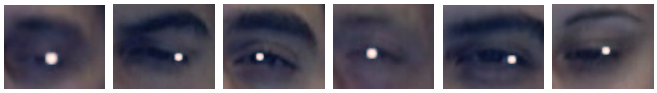


Fig. 16. First frames in which the eye center estimations are off by more than 5% of the interocular distance.

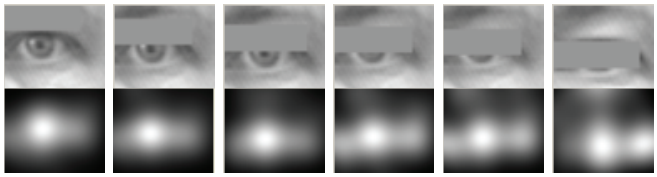


Fig. 17. The effect of eye occlusion on the centermap. Other dark features gain more relevance as the eye's circular pattern gets occluded.

21 subjects followed a moving dot on a computer screen was performed. In the experiment, the dot crosses key locations, in which the frames are saved and manually annotated for the eye location. The key locations are defined by the pixel value in which the dot is displayed on the screen in 6x4 key locations, starting at 50x50 pixels and ending at 1200x740 pixels, in increments of 230 pixels on the horizontal and vertical direction, respectively. Given the size of the screen (40 inches) and the distance of the subjects (750mm), this value indicates a horizontal and an approximate vertical span of $46°$ and $24°$, respectively. The subjects were requested to keep the head as static as possible while following the dot. However, we noted that every subject performed some slight head movements to be able to comfortably gaze at the dot moving at peripheral locations of the screen. This indicates that the subjects were not comfortable to reach the peripheral key location without moving their head. Therefore, we can argue that these peripheral locations reached the limit of 'natural' eye rotation. Since the built dataset was free of occlusions (besides the occlusion caused by the eyelids when the eye is significantly rotated), the achieved accuracy for $e \leq 0.05$ was 100% in all key locations. This result proves that the proposed method is not significantly affected by natural eye rotations,

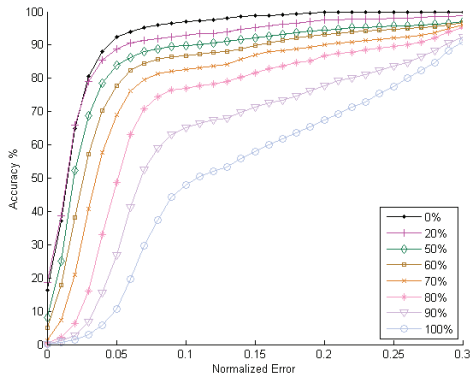


Fig. 18. The effect of occluding eyes at different percentages on the BioID database.

	Horizontal (pixels)					
Vertical (pixels)	50	280	510	740	970	1200
50	76.19%	80.95%	100%	71.43%	85.71%	80.95%
280	90.48%	85.71%	95.24%	100%	85.71%	100%
510	90.48%	85.71%	80.95%	100%	76.19%	71.43%
740	95.24%	71.43%	76.19%	71.43%	76.19%	66.67%

TABLE 7
Effect of changes in eye rotation for $e \leq 0.02$, using MIC+MS.

and therefore it is not affected by the natural occlusions from the eyelids in extreme locations and by the change in shape of the iris due to the rotation. Table 7, shows the average accuracy at the selected key locations for $e \leq 0.02$. At this extremely small range, errors start to be significant when moving away from the central area. Note that in some peripheral areas the accuracy is still 100%. We believe that this is due to the head movements required to gaze at the moving dot comfortably.

4.7 Discussion

As stated in the introduction, the accuracy of the proposed system should not be compared to commercial eye-gaze trackers. The approach discussed here is targeted to niche applications where eye location information is useful but constrained on low resolution imagery, for applications in which close up view or corneal reflection is unavailable (*e.g.* facebook images) and where the use of an eye-gaze tracker would be prohibitively expensive or impractical (*e.g.* automatic red eye reduction on a picture camera).

One of the advantages of the proposed approach that should be discussed is its low computational complexity, since the basic system (without scale space and classification) only requires the computation of image derivatives which is linear in the size of the image and the scale ($O(\sigma N)$). This allows for a real-time implementation while keeping a competitive accuracy with respect to the state of the art. On a 2.4GHz Intel Core 2 Duo, using a single core implementation, the system was able to process ≈ 2500 eye regions per second on a 320x240 image. Including the face detector and the mean shift procedure, the algorithm takes 11ms per frame, which roughly corresponds to 90 frames per second. Therefore, the final frame rate is only limited by the web cam's frame rate. By using the scale space approach, the accuracy improved by about 2% and the system benefits from improved independence to scale conditions. In this way, the method can be applied to different situation without needing an ad-hoc parameter search. In our settings, the scale space MICs+SIFT variant still achieves real time performance (≈ 29 frames per second).

Depending on the target application, a tradeoff between the discussed increase in accuracy and the computational complexity must be chosen. For instance, even if the best results are obtained by the MICs+SIFT method, applying it to video frames thirty times per second will necessarily result in unstable estimates. However, the MIC+MS method scales perfectly to use temporal information: the converged position of the MS window can be kept as initialization for the next frame, and the eye locator can be used to reinitialize the tracking procedure when it is found to be invalid (*i.e.* when the current MIC falls outside the mean shift window). This synergy between the two components allows the tracking system to be fast, fully autonomous and user independent, which is preferable to the less stable, data dependent but more accurate MICs+SIFT variant.

Given the high accuracy and low computational requirements, we foresee the proposed method to be successfully adopted as a preprocessing step to other systems. In particular, systems using classifiers (*e.g.* [7], [27], [42]) should benefit from the reduction in the search and learning phases and can focus on how to discriminate between few candidates. Furthermore, note that our system does not involve any heuristics or prior knowledge to discriminate between candidates. We therefore suggest that it is possible to achieve superior accuracy by integrating the discussed method into systems using contextual information (*e.g.* [13], [24]).

5 Conclusions

In this paper, a new method to infer eye center location using circular symmetry based on isophote properties is proposed. For every pixel, the center of the osculating circle of the isophote is computed from smoothed derivatives of the image brightness, so that each pixel can provide a vote for its own center. The use of isophotes yields low computational cost (which allows for real-time processing) and robustness to rotation and linear illumination changes. A scale space framework is used to improve the accuracy of the proposed method and to gain robustness to scale changes.

An extensive evaluation of the proposed approach was performed, testing it for accurate eye location in standard low resolution images and for robustness to illumination, pose, occlusion, eye rotation, resolution, and scale changes. The comparison with the state of the art suggested that our method is able to achieve highest accuracy and can be successfully applied do very low resolution image of eyes, but this is somewhat bounded by the presence of at least 40% of the circular eye pattern in the image. Given the reported accuracy of the system, we believe that the proposed method provides enabling technology to niche applications in which a good estimation of the eye center location at low resolutions is fundamental.

REFERENCES

[1] M. Asadifard and J. Shanbezadeh. Automatic adaptive center of pupil detection using face detection and cdf analysis. In *IMECS*, 2010.

[2] S. Asteriadis, N. Nikolaidis, A. Hajdu, and I. Pitas. An eye detection algorithm using pixel to edge information. In *Int. Symp. on Control, Commun. and Sign. Proc.*, 2006.

[3] L. Bai, L. Shen, and Y. Wang. A novel eye location algorithm based on radial symmetry transform. In *ICPR*, pages 511–514, 2006.

[4] R. Bates, H. Istance, L. Oosthuizen, and P. Majaranta. Survey of de-facto standards in eye tracking. In *COGAIN Conf. on Comm. by Gaze Inter.*, 2005.

[5] BioID Technology Research. The BioID Face Database. http://www.bioid.com, 2001.

[6] M. Bohme, A. Meyer, T. Martinetz, and E. Barth. Remote eye tracking: State of the art and directions for future development. In *Conf. on Communication by Gaze Interaction*, 2006.

[7] P. Campadelli, R. Lanzarotti, and G. Lipori. Precise eye localization through a general-to-specific model definition. In *BMVC*, 2006.

[8] COGAIN. Communication by gaze interaction: Gazing into the future. http://www.cogain.org, 2006.

[9] C. Colombo, D. Comanducci, and A. del Bimbo. Robust tracking and remapping of eye appearance with passive computer vision. *TOMCCAP*, 3(4), 2007.

[10] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *PAMI*, 25(5):564–577, 2003.

[11] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *PAMI*, 23(6):681–685, 2001.

[12] D. Cristinacce and T. Cootes. Feature detection and tracking with constrained local models. In *BMVC*, 2006.

[13] D. Cristinacce, T. Cootes, and I. Scott. A multi-stage approach to facial feature detection. In *BMVC*, pages 277–286, 2004.

[14] E. B. Dam and B. ter Haar Romeny. *Front End Vision and Multi-Scale Image Analysis*. Kluwer, 2003.

[15] A. T. Duchowski. *Eye Tracking Methodology: Theory and Practice*. Springer, 2007.

[16] S. Duffner. *Face Image Analysis With Convolutional Neural Networks*. PhD thesis, Albert-Ludwigs-Universitat Freiburg, 2008.

[17] R. P. W. Duin. Prtools version 3.0: A matlab toolbox for pattern recognition. In *SPIE*, 2000.

[18] B. Froba and A. Ernst. Face detection with the modified census transform. *Aut. Face and Gest. Recog.*, pages 91–96, 2004.

[19] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *PAMI*, 23(6):643–660, 2001.

[20] J. Geusebroek, A. Smeulders, and J. van de Weijer. Fast anisotropic gauss filtering. *TIP*, 12, 2002.

[21] G. Ghinea, C. Djeraba, S. Gulliver, and K. P. Coyne. Introduction to the special issue on eye-tracking applications in multimedia systems. *TOMCCAP*, 3(4), 2007.

[22] G. Ghinea, C. Djeraba, S. R. Gulliver, and K. P. Coyne. Introduction to special issue on eye-tracking applications in multimedia systems. *TOMCCAP*, 3(4), 2007.

[23] R. Gross, I. Matthews, J. F. Cohn, T. Kanade, and S. Baker. Multi-pie. In *FG*, 2008.

[24] M. Hamouz, J. Kittlerand, J. K. Kamarainen, P. Paalanen, H. Kalviainen, and J. Matas. Feature-based affine-invariant localization of faces. *PAMI*, 27(9):1490–1495, 2005.

[25] W. R. Hendee and P. N. Wells. *The Perception of Visual Information, 2e*. Springer, 1997.

[26] J. Huang and H. Wechsler. Visual routines for eye location using learning and evolution. *Evolutionary Computation*, 4(1), 2000.

[27] O. Jesorsky, K. J. Kirchbergand, and R. Frischholz. Robust face detection using the Hausdorff distance. In *Audio and Video Biom. Pers. Auth.*, pages 90–95, 1992.

[28] Q. Ji, H. Wechsler, A. Duchowski, and M. Flickner. Special issue: eye detection and tracking. *CVIU*, 98(1), 2005.

[29] C. Kervrann, M. Hoebeke, and A. Trubuil. Isophotes selection and reaction-diffusion model for object boundaries estimation. *IJCV*, 50:63–94, 2002.

[30] S. Kim, S.-T. Chung, S. Jung, D. Oh, J. Kim, and S. Cho. Multi-scale gabor feature based eye localization. In *World Academy of Science, Engineering and Technology*, 2007.

[31] J. Koenderink and A. J. van Doorn. Surface shape and curvature scales. *Image and Vision Computing*, pages 557–565, 1992.

[32] B. Kroon, A. Hanjalic, and S. M. Maas. Eye localization for face matching: is it always useful and under what conditions? In *CIVR*, 2008.

[33] J. Lichtenauer, E. Hendriks, and M. Reinders. Isophote properties as features for object detection. In *CVPR*, volume 2, pages 649–654, 2005.

[34] D. Lowe. Distinctive image features from scale-invariant keypoints. In *IJCV*, volume 20, pages 91–110, 2003.

[35] C. H. Morimoto and M. R. M. Mimica. Eye gaze tracking techniques for interactive applications. *CVIU*, 98(1), April 2005.

[36] Z. Niu, S. Shan, S. Yan, X. Chen, and W. Gao. 2D cascaded adaboost for eye localization. In *ICPR*, 2006.

[37] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss. The FERET evaluation methodology for face recognition algorithms. *PAMI*, 22:1090–1104, 2000.

[38] M. Reale, S. Canavan, L. Yin, K. Hu, and T. Hung. A multi-gesture interaction system using a 3d iris disk model for gaze estimation and an active appearance model for 3d hand pointing. *IEEE Transactions on Multimedia*, 13(3), 2011.

[39] D. Reisfeld, H. Wolfson, and Y. Yeshurun. Context free attentional operators: the generalized symmetry transform. *IJCV*, 14:119–130, 1995.

[40] D. Stavens and S. Thrun. Unsupervised learning of invariant features using video. In *CVPR*, 2010.

[41] F. Timm and E. Barth. Accurate eye centre localisation by means of gradients. In *VISAPP*, 2011.

[42] M. Türkan, M. Pardás, and A. Çetin. Human eye localization using edge projection. In *Comp. Vis. Theory and App.*, 2007.

[43] R. Valenti and T. Gevers. Accurate eye center location and tracking using isophote curvature. In *CVPR*, 2008.

[44] M. van Ginkel, J. van de Weijer, L. van Vliet, and P. Verbeek. Curvature estimation from orientation fields. In *SCIA*, 1999.

[45] P. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, 2004.

[46] J. Wang, L. Yin, and J. Moore. Using geometric properties of topographic manifold to detect and track eyes for human-computer interaction. *TOMCCAP*, 3(4), 2007.

[47] P. Wang, M. B. Green, Q. Ji, and J. Wayman. Automatic eye detection and its validation. In *IEEE Workshop on Face Recognition Grand Challenge Experiments*, page 164, 2005.

[48] P. Wang and Q. Ji. Multi-view face and eye detection using discriminant features. *CVIU*, 105(2), 2007.

[49] Z. H. Zhou and X. Geng. Projection functions for eye detection. In *Pattern Recognition*, pages 1049–1056, 2004.

[50] Z. Zhu and Q. Ji. Robust real-time eye detection and tracking under variable lighting conditions and various face orientations. *CVIU*, 98(1):124–154, 2005.

Roberto Valenti received his M.Sc degree with high honors at the University of Amsterdam, The Netherlands. He is currently completing his Ph.D. at the Intelligent Systems Lab Amsterdam at the University of Amsterdam. His research mainly focuses on sensing and understanding users' interactive actions and intentions, multimodal and affective human-computer interaction, the estimation of the human visual gaze and behavior analysis. He is a co-founder and chief technology officer of ThirdSight, a spin-off of the University of Amsterdam, focused on the automatic analysis of faces. He is a member of the IEEE.

Theo Gevers is an associate professor of computer science with the University of Amsterdam, The Netherlands, and a full professor at the Computer Vision Center, Universitat Autònoma de Barcelona, Spain. At the University of Amsterdam, Theo Gevers is a teaching director of the MSc in Artificial Intelligence. He currently holds a VICI Award (for research excellence) from the Dutch Organisation for Scientific Research. He is a co-founder and chief scientific officer of ThirdSight, a spin-off of the UvA. His main research interests are in the fundamentals of image understanding, object recognition and color in computer vision. Further, he is interested in different aspects of human behavior, specifically in emotion recognition. He is the chair for various conferences and is an associate editor for the IEEE Transactions on Image Processing. Further, he is a program committee member for a number of conferences, and an invited speaker at major conferences. He is a lecturer delivering postdoctoral courses given at various major conferences (CVPR, ICPR, SPIE, and CGIV). He is a member of the IEEE.