14th European Signal Processing Conference (EUSIPCO 2006), Florence, Italy, September 4-8, 2006, copyright by EURASIP

A NOVEL EYE DETECTION ALGORITHM UTILIZING EDGE-RELATED GEOMETRICAL INFORMATION

S. Asteriadis, N. Nikolaidis, A. Hajdu, I. Pitas

Department of Informatics, Aristotle University of Thessaloniki,
BOX 451, 54124, Thessaloniki, Greece,
phone: +(30)2310996361, fax: +(30)2310998453, pitas@aiia.csd.auth.gr, http://poseidon.csd.auth.gr/

ABSTRACT

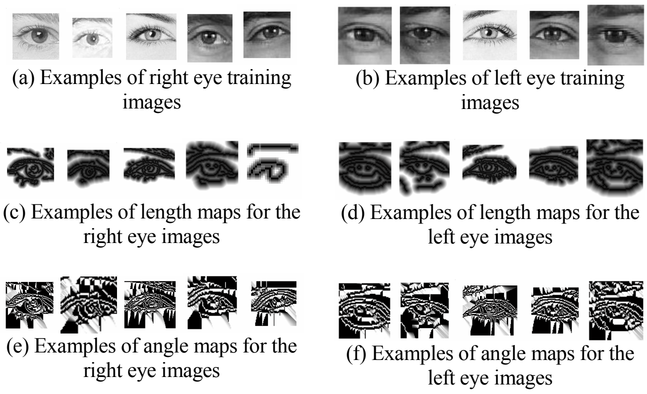
A novel method for eye detection and eye center localization, based on geometrical information is described in this paper. First, a face detector is applied to detect the facial region, and the edge map of this region is extracted. A vector pointing to the closest edge pixel is then assigned to every pixel. Length and slope information for these vectors is used to detect the eyes. For eye center localization, intensity information is used. The proposed method can work on low-resolution images and has been tested on two face databases with very good results.

1. INTRODUCTION

Eye detection and eye center localization has been a topic of intense research, leading into a multitude of different approaches. In [1] the authors use Generalized Projection Functions (GPF) to locate the eye centers in an area found using the algorithm proposed in [2]. In [3], a three stage approach is proposed: the face is initially localized using the Hausdorff distance [4]. Then, a closer search of the area of the eyes is performed in the same way on the detected face and, finally the Multi-Layer Perceptron (MLP) is applied to provide a more accurate pupil localization. A multi-stage approach to detect 17 salient facial features, including the eyes, is proposed in [5]. First, a face detector is applied, and subsequently the Pairwise Reinforcement of Feature Responses (PRFR) is used to detect the salient features. A version of the Active Appearance Model (AAM) search is finally used for refinement.

A novel algorithm for eye detection on face images, including low resolution ones, is presented in this paper. Pixel intensity information which is customarily used in eye localization approaches might prove unreliable due to varying or unfavorable lighting conditions. Moreover, in low resolution images, details of the eyes sometimes are lost. An approach which detects the eye region, based on geometric information from the eye and the surrounding area is proposed in this paper. The method can work very well even in low resolution images. Experiments involving the XM2VTS [6] and BioID [7] databases show that the method can achieve performance that is comparable to that of methods demanding higher resolution images.

The organization of the paper is as follows: Eye region detection and eye center localization are described in Sections 2 and 3 respectively. Experimental results are provided in Section 4. Conclusions follow.



(a) Examples of right eye training images

(b) Examples of left eye training images

(c) Examples of length maps for the right eye images

(d) Examples of length maps for the left eye images

(e) Examples of angle maps for the right eye images

(f) Examples of angle maps for the left eye images

Figure 1: Training data

2. DETECTION OF EYE REGIONS

The outline of the proposed eye region detection method is as follows: First, the Canny edge detector [8] is applied on the output of a face detector. This edge detector has been selected because it can be adjusted so that it finds edges of different intensity. Then, for each pixel, the vector that points to the closest edge pixel is calculated. The magnitude (length) and the slope (angle) of each vector are assigned to the corresponding pixel. Thus, instead of the intensity values of each pixel, we generate and use in the proposed algorithm the vector length and angle maps of the image. PCA is subsequently applied on a set of training eye images to derive eigenvectors for these maps. In order to detect the eye regions on an image, the length and angle maps of candidate regions are evaluated and are projected on the spaces spanned by the eigenvectors found during training. The similarity of the candidate regions in the new feature spaces with the features derived for a pair of prototype eye models is used to declare an eye presence. The steps of the algorithm are described in detail in the paragraphs below.

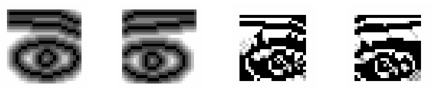
For the extraction of the eigenvectors in the training part of the algorithm k=58 right and left eye regions, collected from face images found on the web, were used (Figures 1a, b). The training images were scaled to NxM pixels (N=M=25) and the Canny edge detector was applied. The

(a) (b) (c) (d)

Figure 2: (a) Detected face, (b) thresholded Canny edge detector output (c) vector magnitude map (d) vector slope map

(a) (b)

Figure 3: (a) Right model-eye, (b) left model-eye



(a) (b)

Figure 4: (a):Length maps of model eyes, (b): angle maps of model eyes.

corresponding vector length (Figures 1c,d) and vector angle maps (Figures 1e,f) for each training image form two matrices of dimension *N*x*M*. These matrices can be alternatively considered as two 1-D vectors of dimension $L=NM$. A normalization step that consists in subtracting, from each matrix, the respective average matrix follows. Then, PCA is applied separately on the length and angle maps of the left and right eye training images resulting in 4x58 eigenvectors $U_{Ra,i}$, $U_{Rl,i}$, $U_{La,i}$, $U_{Ll,i}$ ($1<i<58$), each being of dimension *1xNM,* that correspond to the angle and length maps of the right and left eye. Despite the fact that the angular data are of periodic nature and thus the use of the standard PCA on them is not very well grounded, the obtained results presented in section IV are very good, especially when compared to using only length information.
Prior to the actual eye detection, a face detector is applied on the image. In our case, the face detector proposed in [9, 10, 11] was utilized. The detected face area was then scaled to 150x90 pixels since experiments proved that images of such dimensions retain all necessary information for detecting the eyes with the proposed technique.
Subsequently, edge detection was performed. Since edges related to eyes are among the most prominent edges in a face, parameter values of the Canny edge detector that detect the most significant edges, were used (Figure 2b). Values equal to 50 and 25 were used for the high and low threshold respectively, whereas sigma was set to 1 [8].
Vector length and angle maps of the face can be seen in Figures 2c, d respectively.
For the actual eye detection, all the areas of size *N*x*M* are examined and the vectors containing the weights (features) that project the normalized length and angle maps Φ_{Ra}, Φ_{Rl}, Φ_{La}, Φ_{Ll} (represented as 1-D vectors) of a certain area on the corresponding feature spaces spanned by the eigenvectors evaluated during the training stage are calculated. More specifically, the weight vector elements $w_{Ra,i}$, $w_{Rl,i}$, $w_{La,i}$, $w_{Ll,i}$ for a given *NxM* area are calculated using the formulas below:

$$w_{Ra,i} = U_{Ra,i}{}^{T} \cdot \Phi_{Ra} \quad (1\text{-a})$$

$$w_{Rl,i} = U_{Rl,i}{}^{T} \cdot \Phi_{Rl} \quad (1\text{-b})$$

$$w_{La,i} = U_{La,i}{}^{T} \cdot \Phi_{La} \quad (1\text{-c})$$

$$w_{Ll,i} = U_{Ll,i}{}^{T} \cdot \Phi_{Ll} \quad (1\text{-d})$$

Thus, each area is represented by two *k*-dimensional vectors, composed by the weights needed to project the *NM*-dimensional vectors of angles and lengths on the respective *k*-dimensional space (in our case, k=58).
To proceed with the detection, prototype eye models of dimensions *25*x*25* were used (Figure 3). For each of these two models, the procedure described above is followed. In other words, the vector length map and vector angle map are derived (Figure 4), and the respective projection weights $w_{Ra,Model}$, $w_{Rl,Model}$, $w_{La,Model}$, $w_{Ll,Model}$ are calculated. For each candidate area on the face, the projection weights w_{Rai}, w_{Rli}, w_{Lai}, w_{Li}, are compared to those of the model eyes using the L_2 norm:

$$L_{2,R} = \| w_{Ra} - w_{Ra,Model} \| + \| w_{Rl} - w_{Rl,Model} \| \quad (2\text{-a})$$

$$L_{2,L} = \| w_{La} - w_{La,Model} \| + \| w_{Ll} - w_{Ll,Model} \| \quad (2\text{-b})$$

The candidate areas with the smallest distance from the model eyes are declared as eye areas. To speed up the algorithm, knowledge of the approximate positions of eyes on a face is used. More specifically, candidate eye regions are searched only in a zone on the upper part of the detected face and the right and left eyes are searched at the right and left parts of this zone, respectively.

3. EYE CENTER LOCALIZATION

Before eye centre localization, a pre-processing step needs to be taken. Since reflections (highlights), that affect the results in a negative way, frequently appear on the eye, a reflection removal step is implemented. Such highlights are usually bright areas consisting of no more than a few pixels. Thus, the reflection removal step proceeds as follows: The eye area (Figure 5a) is first converted into a binary image (Figure 5b) through thresholding. This is done using Otsu's method [12], which chooses the threshold so as to minimize the intraclass variance of the thresholded black and white pixels. Subsequently, all the white connected components that have fewer than 8 pixels are considered as highlight areas (see figure 5c for the binary image having these areas removed) and the intensities of the pixels in the grayscale image that correspond to these areas are substituted by the average of their surrounding pixels. The result is an eye area with most highlights removed (Figure 5d)

By inspecting the eye images used for training, one can observe that the eyes usually reside at the lower part of the detected area. Experiments have shown that the region comprising of the lower 15x25 pixels covers the very area of the eye, whereas the upper region covers the eyebrow. Thus, the information in this area comes from the eye itself and not from the eyebrow or the glasses and searching for the eye centers is limited to this area only.

Since, at the actual eye center position, there is a lot of luminance variation along the horizontal axis, the image $D_x(x,y)$ of the discrete intensity derivatives along the horizontal direction is evaluated:

$$D_x(x,y) = I(x,y) - I(x-1,y) \qquad (3)$$

The contents of this derivative image are subsequently projected on the vertical axis:

$$D_x^p(y) = \sum_x D_x(x,y) \qquad (4)$$

and the 3 horizontal lines, corresponding to the 3 largest projections are selected. The middle of these lines defines the vertical position of the eye center (Figure 6a). Exactly the same procedure (involving in this case the derivatives in the vertical direction) is used to find the horizontal position of the eye center (Figure 6b).

This result can be further refined by applying the same localization method within an area of 15x25 pixels centered around the point found in the previous step. For further refinement, the fact that the iris of the eye is the darkest area near the point found at the previous step can be used in order to place the eye centre exactly in the middle of it. This is done by using the found eye centre and a search area of ±4 pixels around it and looking for the darkest 5x5 subarea. The refined eye center is located in the centre of this sub-area.

(a) (b) (c) (d)

Figure 5: Light reflections removal steps

(a) (b)

Figure 6: Horizontal and vertical discrete derivative images

Table I: RESULTS ON THE XM2VTS DATABASE

	Eye regions	Eye centers
People without glasses	99.7%	99.7%
People with glasses	96.2%	95.0%
Total	98.5%	98%

Table II: RESULTS ON THE BioID DATABASE

	Eye regions	Eye centers
People without glasses	99.5%	99.1%
People with glasses	98.3%	98.2%
People with closed eyes	100%	100%
Total	99.1%	98.8%

(a)

(b)

Figure 7: Examples of BioID (a) and XM2VTS (b) databases

4. EXPERIMENTAL PERFORMANCE EVALUATION

The method has been tested on two databases, namely the BioID [7] and the extended M2VTS [6]. The BioID database contains 1521 frontal facial images acquired under various lighting conditions, in a complex background. Furthermore, the BioID database contains tilted and rotated faces. Thus, this database is considered as the most challenging one for the eye detection task. Since, in the BioID database, many faces lie close to the upper and lower image borders, two gray bands were added above and below the images, prior to face detection. A similar preprocessing step was applied in [5]. Furthermore, the proposed method was applied on 600 images randomly selected from the XM2VTS database. The images in this database were acquired under controlled lighting conditions in a uniform background, a fact that facilitates the face detection step. It should be noted that both databases contain persons that wear glasses. Furthermore, in a few images of the BioID database, people have their eyes shut or pose various expressions. A few sample images from the two databases can be seen in Figure 7.

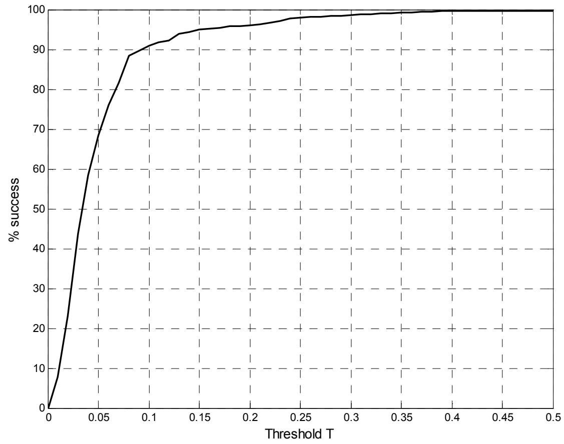
Two types of results were obtained on the images described above: results regarding eye region detection and results on eye center localization. For eye region detection, visual inspection of the detected regions was used to declare success or failure. Correct detection percentages are listed in the columns of Tables I and II denoted as “Eye regions”. It is obvious that the obtained detection rates are very good. One can notice that the proposed eye region detection method works perfectly in the few images of people having their eyes shut. This can be attributed to the fact that the length/angle maps of regions related to shut eyes are similar to the ones of an open eye.

To evaluate the effectiveness of utilizing geometrical information (angle and length maps) instead of intensity information, we compared the eye detection results with those obtained by applying PCA on the pixel intensity. The overall correct detection results using pixel intensity were 92.3% on the BioID database, that is 6.8% lower than in the case of using length and angle maps with the same parameters (where applicable).

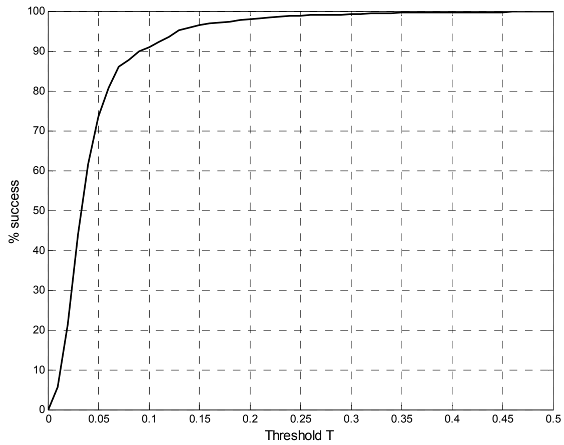
For the eye center localization, the correct detection rates were calculated through the following criterion, used in [3]:

$$m_e = \frac{\max(d_1, d_2)}{s} < T$$

In the previous formula, d_1 and d_2 are the distances between the manually labeled eye centers (ground truth) and the eye centers found by the algorithm, s is the distance between the two manually labeled eye centers and T is the threshold used to declare a successful detection. The correct detection rates for T=0.25 are presented in Tables I and II (column “Eye centers”). Furthermore, the success rates on both databases for various values of the threshold T are depicted in Figure 8. Results in Tables I, II show that glasses do not introduce significant error when compared to cases where subjects are not wearing glasses. This is because the basic shape of the eye region remains unchanged, even if the eyebrows are occluded because of the glasses skeleton. In such cases, the geometry of the eyebrows is replaced by the upper part of the skeleton of the glasses. Detailed inspection of the results revealed that most failures occur when subjects are wearing glasses in combination to having their eyes shut. Moreover, some failures occur when subjects have long hair on the forehead, resulting in wrong eye area detection, as edges stemming from hair might introduce false alarms. Some detection and eye center localization results are depicted in Figure 9.



(a)



(b)

Figure 8: Eye center localization results on (a): XM2VTS and (b) BioID

Despite the fact that the test images were subsampled by a factor of, sometimes, more than 2, the results are comparable to those achieved by other state-of-the-art methods. Since the face detector had a success of 98.8% on the BioID database, the overall result for $T=0.25$ is 0.988x0.988=0.976 i.e., 97.6%, while the method in [3] achieved 91.8% and the method in [1] had a success rate of 94.81%. For T=0.1 i.e. when the correct eye localization criterion is more strict, the proposed method reaches 91.1% on BioID (90% when the face detector is taken into account) while the method in [3] achieves 79%. However, in this case the method proposed in [5] outperforms the proposed method with a correct detection rate of 96%.

5. CONCLUSIONS

A novel method for eye detection was proposed in this paper. The method utilizes geometrical characteristics of the eye and the surrounding area. The method proved to perform very well even on facial images of relatively small dimensions taken from challenging databases such as the BioID database. The results are in most of the cases better than those achieved by other state-of the-art methods.

ACKNOWLEDGMENT

This work has been partially supported by the FP6 European Union Network of Excellence MUSCLE "Multimedia Understanding Through Semantic Computation and Learning" (FP6-507752).

Figure 9: Some eye localization results. The last two images in the last row represent incorrect eye localization cases.

REFERENCES

[1] Z.H. Zhou and X. Geng, "Projection functions for eye detection", *Pattern Recognition 37*, no 5, pp. 1049-1056, 2004
[2] J. Wu, Z.-H.Zhou, "Efficient face candidates selector for face detection", *Pattern Recognition 36,* no5, pp. 1175-1186, 2003
[3] O. Jesorsky, K. J. Kirchberg, and R. W. Frischholz, "Robust face detection using the hausdorff distance", *3rd International Conference on Audio and Video-based Biometric Person Authentication*, Halmstad, Sweden, pp. 90-95, 2001
[4] W. Rucklidge, "Efficient Visual Recognition Using the Hausdorff distance", vol 1173 of *Lecture notes in computer science*. Springer, 1996
[5] D. Cristinacce, T. Cootes, and I. Scott, "A multi-stage approach to facial feature detection", *Proc. Of BMVC*, pp. 231-240, London 2004
[6] K. Messer, J. Matas, J. Kittler, J. Luettin and G. Maitre. XM2VTSDB: The extended M2VTS database. *Second International Conference on Audio and Video-based Biometric Person Authentication*, pp. 72-77, March 1999
[7] The BioID face database: [http://www.bioid.com/downloads/facedb/facedatabase.html]
[8] J. Canny "A Computational Approach to Edge Detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol 8, no. 6, 1986
[9] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 1, pp. 511-518, 2001
[10] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," *IEEE ICIP*, vol. 1, pp. 900-903, Rochester, NY, 2002
[11] P. Viola and M. Jones, "Robust real-time face detection" *International Journal of Computer Vision 57*, pp. 137-154, 2004
[12] Otsu, N., "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, No. 1, 1979, pp. 62-66.