HUMAN EYE LOCALIZATION USING EDGE PROJECTIONS

Mehmet Türkan

Dept. of Electrical and Electronics Engineering, Bilkent University, Bilkent, 06800 Ankara, Turkey
turkan@ee.bilkent.edu.tr

Montse Pardàs

Dept. of Signal Theory and Communications, Technical University of Catalonia, 08034 Barcelona, Spain
montse@gps.tsc.upc.es

A. Enis Çetin

Dept. of Electrical and Electronics Engineering, Bilkent University, Bilkent, 06800 Ankara, Turkey
cetin@bilkent.edu.tr

Abstract: In this paper, a human eye localization algorithm in images and video is presented for faces with frontal pose and upright orientation. A given face region is filtered by a high-pass filter of a wavelet transform. In this way, edges of the region are highlighted, and a caricature-like representation is obtained. After analyzing horizontal projections and profiles of edge regions in the high-pass filtered image, the candidate points for each eye are detected. All the candidate points are then classified using a support vector machine based classifier. Locations of each eye are estimated according to the most probable ones among the candidate points. It is experimentally observed that our eye localization method provides promising results for both image and video processing applications.

1 INTRODUCTION

The problem of human eye detection, localization, and tracking has received significant attention during the past several years because of wide range of human-computer interaction and surveillance applications. As eyes are one of the most important salient features of a human face, detecting and localizing them helps researchers working on face detection, face recognition, iris recognition, facial expression analysis, etc.

In recent years, many heuristic and pattern recognition based methods have been proposed to detect and localize eyes in still images and video. Most of these methods described in the literature ranging from very simple algorithms to composite high-level approaches are highly associated with face detection and face recognition. Traditional image-based eye detection methods assume that the eyes appear different from the rest of the face both in shape and intensity. Dark pupil, white sclera, circular iris, eye corners, eye shape, etc. are specific properties of an eye to distinguish it from other objects (Zhu and Ji, 2005). (Morimoto and Mimica, 2005) reviewed the state of the art of eye gaze trackers by comparing the strengths, and weaknesses of the alternatives available today. They

also improved the usability of several remote eye gaze tracking techniques. (Zhou and Geng, 2004) defined a method for detecting eyes with projection functions. After localizing the rough eye positions using (Wu and Zhou, 2003) method, they expand a rectangular area near each rough position. Special cases of generalized projection function (GPF) are used to localize the cental positions of eyes in eye windows.

Recently, wavelet domain (Daubechies, 1990; Mallat, 1989) feature extraction methods have been developed and become very popular (Garcia and Tziritas, 1999; Zhu et al., 2000; Viola and Jones, 2001; Huang and Mariani, 2000) for face and eye detection. (Zhu et al., 2000) described a subspace approach to capture local discriminative features in the space-frequency domain for fast face detection based on orthonormal wavelet packet analysis. They demonstrated the detail (high frequency sub-band) information within local facial areas contain information about eyes, nose and mouth, which exhibit noticeable discrimination ability for face detection problem. This assumption may also be used to detect and localize facial areas such as eyes. (Huang and Mariani, 2000) described a method to represent eye images using wavelets. Their algorithm uses a structural model to characterize the geometric pattern of facial compo-

nents, i.e., eyes, nose, and mouth, using multiscale filters. They perform eyes detection using a neural network classifier. (Cristinacce et al., 2004) developed a multi-stage approach to detect features on a human face, including the eyes. After applying a face detector to find the approximate location of the face in the image, they extract and combine features using Pairwise Reinforcement of Feature Responses (PRFR) algorithm. The estimated features are then refined using a version of the Active Appearance Model (AMM) search which is based on edge and corner features.

In this study, a human eye localization method in images and video is proposed with the assumption that a human face region in a given still image or video frame is already detected by means of a face detector. This method is basically based on the idea that eyes can be detected and localized from edges of a typical human face. In fact, a caricaturist draws a face image in a few strokes by drawing the major edges, such as eyes, nose, mouth, etc., of the face. Most wavelet domain image classification methods are also based on this fact because significant wavelet coefficients are closely related with edges (Mallat, 1989; Cetin and Ansari, 1994; Garcia and Tziritas, 1999).

The proposed algorithm works with edge projections of given face images. After an approximate horizontal level detection, each eye is first localized horizontally using horizontal projections of associated edge regions. Then, horizontal edge profiles are calculated on the estimated horizontal levels. Eye candidate points are determined by pairing up the local maximum point locations in the horizontal profiles with the associated horizontal levels. After obtaining the eye candidate points, verification is carried out by a support vector machine based classifier. The locations of eyes are finally estimated according to the most probable point for each eye separately.

This paper is organized as follows. Section 2 describes our eye localization algorithm where each step is briefly explained for the techniques used in the implementation. In Section 3, experimental results of the proposed algorithm are presented and the detection performance is compared with currently available eye localization methods. Conclusions are given in Section 4.

2 EYE LOCALIZATION SYSTEM

In this paper, a human eye localization scheme for faces with frontal pose and upright orientation is developed. After detecting a human face in a given color image or video frame using edge projections method proposed by (Turkan et al., 2006), the face region is

decomposed into its wavelet domain sub-images. The detail information within local facial areas, e.g., eyes, nose, and mouth, is obtained in low-high, high-low, and high-high sub-images of the face pattern. A brief review of the face detection algorithm is described in Section 2.1, and the wavelet domain processing is presented in Section 2.2. After analyzing horizontal projections and profiles of horizontal-crop and vertical-crop edge images, the candidate points for each eye are detected as explained in Section 2.3. All the candidate points are then classified using a support vector machine based classifier. Finally, the locations of each eye are estimated according to the most probable ones among the candidate points.

2.1 Face Detection Algorithm

After determining all possible face candidate regions using color information in a given still image or video frame, a single-stage 2-D rectangular wavelet transform of each region is computed. In this way, wavelet domain sub-images are obtained. The low-high and high-low sub-images contain horizontal and vertical edges of the region, respectively. The high-high sub-image may contain almost all the edges, if the face candidate region is sharp enough. It is clear that the detail information within local facial areas, e.g., edges due to eyes, nose, and mouth, show noticeable discrimination ability for face detection problem of frontal view faces. (Turkan et al., 2006) take advantage of this fact by characterizing these sub-images using their projections and obtain 1-D projection feature vectors corresponding to edge images of face or face-like regions. Horizontal projection $H[.]$, and vertical projection $V[.]$ are simply computed by summing pixel values, $d[.,.]$, in a row and column, respectively:

$$H[y] = \sum_x |d[x,y]| \qquad (1)$$

and

$$V[x] = \sum_y |d[x,y]| \qquad (2)$$

where $d[x,y]$ is the sum of the absolute values of the three high-band sub-images.

Furthermore, Haar filter-like projections are computed as in (Viola and Jones, 2001) approach as additional feature vectors which are obtained from differences of two sub-regions in the candidate region. The final feature vector for a face candidate region is obtained by concatenating all the horizontal, vertical, and filter-like projections. These feature vectors are then classified using a support vector machine (SVM) based classifier into face or non-face classes.

As wavelet domain processing is used both for face and eye detection, it is described in more detail in the next subsection.

2.2 Wavelet Decomposition of Face Patterns

A given face region is processed using a 2-D filter-bank. The region is first processed row-wise using a 1-D Lagrange filterbank (Kim et al., 1992) with a low-pass and high-pass filter pair, $h[n] = \{0.25, 0.5, 0.25\}$ and $g[n] = \{-0.25, 0.5, -0.25\}$, respectively. Resulting two image signals are processed column-wise once again using the same filterbank. The high-band sub-images that are obtained using a high-pass filter contain edge information, e.g., the low-high and high-low sub-images contain horizontal and vertical edges of the input image, respectively (Fig. 1). The absolute values of low-high, high-low and high-high sub-images can be summed up to have an image having significant edges of the face region.
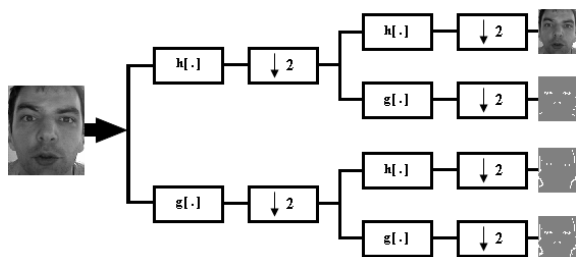

Figure 1: Two-dimensional rectangular wavelet decomposition of a face pattern; low-low, low-high, high-low, high-high sub-images. $h[.]$ and $g[.]$ represent 1-D low-pass and high-pass filters, respectively.

A second approach is to use a 2-D low-pass filter and subtract the low-pass filtered image from the original image. The resulting image also contains the edge information of the original image and it is equivalent to the sum of undecimated low-high, high-low, and high-high sub-images, which we call the detail image as shown in Fig. 2(b).

2.3 Feature Extraction and Eye Localization

The first step of feature extraction is de-noising. The detail image of a given face region is de-noised by soft-thresholding using the method by (Donoho and Johnstone, 1994). The threshold value t_n is obtained as follows:

$$t_n = \sqrt{\frac{2 \log(n)}{n}}\, \sigma \qquad (3)$$

where n is the number of wavelet coefficients in the region and σ is the estimated standard deviation of Gaussian noise over the input signal. The wavelet co-efficients below the threshold are forced to become zero and those above the threshold are kept as are. This initial step removes the noise effectively while preserving the edges in the data.

The second step of the algorithm is to determine the approximate horizontal position of eyes using the horizontal projection in the upper part of the detail image as eyes are located in the upper part of a typical human face (Fig. 2(a)). This provides robustness against the effects of edges due to neck, mouth (teeth), and nose on the horizontal projection. The index of the global maximum in the smoothed horizontal projection in this region indicates the approximate horizontal location of both eyes as shown in Fig. 2(d). By obtaining a rough horizontal position, the detail image is cropped horizontally according to the maximum as shown in Fig. 2(c). Then, vertical-crop edge regions are obtained by cropping the horizontally cropped edge image into two parts vertically as shown in Fig. 2(e).
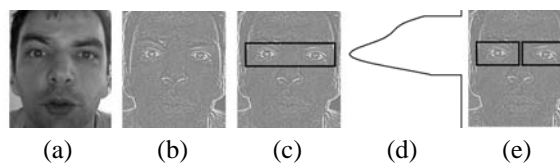

(a)   (b)   (c)   (d)   (e)

Figure 2: (a) An example face region with its (b) detail image, and (c) horizontal-crop edge image covering eyes region determined according to (d) smoothed horizontal projection in the upper part of the detail image (the projection data is filtered with a narrow-band low-pass filter to obtain the smooth projection plot). Vertical-crop edge regions are obtained by cropping the horizontal-crop edge image vertically as shown in (e).

The third step is to compute again horizontal projections in both right-eye and left-eye vertical-crop edge regions in order to detect the exact horizontal positions of each eye separately. The global maximum of these horizontal projections for each eye provides the estimated horizontal levels. This approach of dividing the image into two vertical-crop regions provides some freedom on detecting eyes in oriented face regions where eyes are not located on the same horizontal level.

Since a typical human eye consists of white sclera around dark pupil, the transition from white to dark (or dark to white) area produces significant
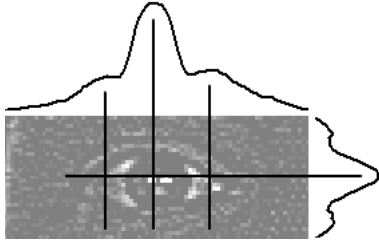
Figure 3: An example vertical-crop edge region with its smoothed horizontal projection and profile. Eye candidate points are obtained by pairing up the maximums in the horizontal profile with the associated horizontal level.

jumps in the coefficients of the detail image. We take advantage of this fact by calculating horizontal profiles on the estimated horizontal levels for each eye. The jump locations are estimated from smoothed horizontal profile curves. An example vertical-crop edge region with its smoothed horizontal projection and profile are shown in Fig. 3. It is worth mentioning that, the global maximum in the smoothed horizontal profile signals is due to the transition both from white sclera to pupil and pupil to white sclera region. The first and last peaks correspond to outer and inner eye corners. Since there is a transition from skin to white sclera (or white sclera to skin) region, these peak values are small compared to those of white sclera to pupil (or pupil to white sclera) region. However, this may not be the case in some eye regions. There may be more (or less) than three peaks depending on the sharpness of the vertical-crop eye region and eye glasses.

Eye candidate points are obtained by pairing up the indices of the maximums in the smoothed horizontal profiles with the associated horizontal levels for each eye. An example horizontal level estimate with its candidate vertical positions are shown in Fig. 3.

An SVM based classifier is used to discriminate the possible eye candidate locations. A rectangle with center around each candidate point is cropped and fed to the SVM classifier. The size of rectangles depends on the resolution of the detail image. However, the cropped rectangular region is finally resized to a resolution of 25x25 pixels. The feature vectors for each eye candidate region are calculated similar to the face detection algorithm by concatenating the horizontal and vertical projections of the rectangles around eye candidate locations. The points that are classified as an eye by SVM classifier are then ranked with respect to their estimated probability values (Wu et al., 2004) produced also by the classifier. The locations of eyes are finally determined according to the most probable point for each eye separately.

In this paper, we used a library for SVMs called LIBSVM (Chang and Lin, 2001). Our simulations are carried out in C++ environment with interface for Python using radial basis function (RBF) as kernel. LIBSVM package provides the necessary quadratic programming routines to carry out the classification. It performs cross validation on the feature set and also normalizes each feature by linearly scaling it to the range $[-1, +1]$. This package also contains a multi-class probability estimation algorithm proposed by (Wu et al., 2004).

3   EXPERIMENTAL RESULTS

The proposed eye localization algorithm is evaluated on the CVL [http://www.lrv.fri.uni-lj.si/] and BioID [http://www.bioid.com/] Face Databases in this paper. All the images in these databases are with head-and-shoulder faces.

The CVL database contains 797 color images of 114 persons. Each person has 7 different images of size 640x480 pixels: far left side view, $45°$ angle side view, serious expression frontal view, $135°$ angle side view, far right side view, smile -showing no teeth- frontal view, and smile -showing teeth- frontal view. Since the developed algorithm can only be applied to faces with frontal pose and upright orientation, our experimental dataset contains 335 frontal view face images from this database. Face detection is carried out using (Turkan et al., 2006) method for this dataset since the images are color.

The BioID database consists of 1521 gray level images of 23 persons with a resolution of 384x286 pixels. All images in this database are the frontal view faces with a large variety of illumination conditions and face size. Face detection is carried out using Intel's OpenCV face detection method[1] since all images are gray level.

The estimated eye locations are compared with the exact eye center locations based on a relative error measure proposed by (Jesorsky et al., 1992). Let C_r and C_l be the exact eye center locations, and $\tilde{C}_r$ and $\tilde{C}_l$ be the estimated eye positions. The *relative error* of this estimation is measured according to the formula:

$$d = \frac{\max\left(\| C_r - \tilde{C}_r \|, \| C_l - \tilde{C}_l \|\right)}{\| C_r - C_l \|} \qquad (4)$$

In a typical human face, the width of a single eye roughly equals to the distance between inner eye

---

[1]More information is available on `http://www.intel.com/technology/itj/2005/volume09issue02/art03_learning_vision/p04_face_detection.htm`

Table 1: Eye localization results.

Method	Database	Success Rate (d<0.25)	Success Rate (d<0.10)
Our method (edge projections)	CVL	99.70%	80.90%
Our method (edge projections)	BioID	99.46%	73.68%
(Asteriadis et al., 2006)	BioID	97.40%	81.70%
(Zhou and Geng, 2004)	BioID	94.81%	–
(Jesorsky et al., 1992)	BioID	91.80%	79.00%
(Cristinacce et al., 2004)	BioID	98.00%	96.00%

corners. Therefore, half an eye width approximately equals to a relative error of 0.25. Thus, in this paper we considered a relative error of $d < 0.25$ to be a correct estimation of eye positions.

Our method has a 99.46% overall success rate for $d < 0.25$ on the BioID database while (Jesorsky et al., 1992) achieved 91.80% and (Zhou and Geng, 2004) had a success rate 94.81%. (Asteriadis et al., 2006) also reported a success rate 97.40% using the same face detector on this database. (Cristinacce et al., 2004) had a success rate 98.00% (we obtained this value from their distribution function of relative eye distance graph). However, our method reaches 73.68% for $d < 0.10$ while (Jesorsky et al., 1992) had 79.00%, (Asteriadis et al., 2006) achieved 81.70%, and (Cristinacce et al., 2004) reported a success rate 96.00% for this strict d value. All the experimental results are given in Table 1 and the distribution function of relative eye distances on the BioID database is shown in Fig. 4. Some examples of estimated eye locations are shown in Fig. 5.
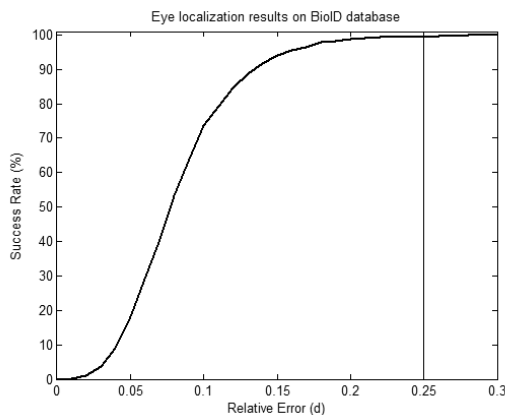

Figure 4: Distribution function of relative eye distances of our algorithm on the BioID database.

The proposed human eye localization system with face detection algorithm (Turkan et al., 2006) is also implemented in .NET C++ environment, and it works in real-time with approximately 12 fps on a standard personal computer.

4 CONCLUSION

In this paper, we presented a human eye localization algorithm for faces with frontal pose and upright orientation. The performance of the algorithm has been examined on two face databases by comparing the estimated eye positions with the ground-truth values using a relative error measure. The localization results show that our algorithm is robust against both illumination and scale changes since the BioID database contains images with a large variety of illumination conditions and face size. To the best of our knowledge, our algorithm gives the best results on the BioID database for $d < 0.25$. Therefore, it can be applied to human-computer interaction applications, and be used as the initialization stage of eye trackers. In eye tracking applications, e.g., (Bagci et al., 2004), a good initial estimate is satisfactory as the tracker further localizes the positions of eyes. For this reason, $d < 0.25$ results are more important than those of $d < 0.10$ from the tracker point of view.

Our algorithm provides approximately 1.5% improvement over the other methods. This may not look that great at first glance but it is a significant improvement in a commercial application as it corresponds to one more satisfied customer in a group of hundred users.

ACKNOWLEDGEMENTS

(a)

(b)

Figure 5: Examples of estimated eye locations from the (a) BioID, (b) CVL database.

REFERENCES

Asteriadis, S., Nikolaidis, N., Hajdu, A., and Pitas, I. (2006). An eye detection algorithm using pixel to edge information. In *Proc. Int. Symposium on Control, Communications, and Signal Processing*. IEEE.

Bagci, A. M., Ansari, R., Khokhar, A., and Cetin, A. E. (2004). Eye tracking using markov models. In *Proc. Int. Conf. on Pattern Recognition*, volume III, pages 818–821. IEEE.

Cetin, A. E. and Ansari, R. (1994). Signal recovery from wavelet transform maxima. *IEEE Trans. on Signal Processing*, 42:194–196.

Chang, C. C. and Lin, C. J. (2001). *LIBSVM: a library for support vector machines*. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Cristinacce, D., Cootes, T., and Scott, I. (2004). A multi-stage approach to facial feature detection. In *Proc. BMVC*, pages 231–240.

Daubechies, I. (1990). The wavelet transform, time-frequency localization and signal analysis. *IEEE Trans. on Information Theory*, 36:961–1005.

Donoho, D. L. and Johnstone, I. M. (1994). Ideal spatial adaptation via wavelet shrinkage. *Biometrika*, 81:425–455.

Garcia, C. and Tziritas, G. (1999). Face detection using quantized skin color regions merging and wavelet packet analysis. *IEEE Trans. on Multimedia*, 1:264–277.

Huang, W. and Mariani, R. (2000). Face detection and precise eyes location. In *Proc. Int. Conf. on Pattern Recognition*, volume IV, pages 722–727. IEEE.

Jesorsky, O., Kirchberg, K. J., and Frischholz, R. W. (1992). Robust face detection using the hausdorff distance. In *Proc. Int. Conf. on Audio- and Video-based Biometric Person Authentication*, volume 2091, pages 90–95. Springer, Lecture Notes in Computer Science.

Kim, C. W., Ansari, R., and Cetin, A. E. (1992). A class of linear-phase regular biorthogonal wavelets. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, volume IV, pages 673–676. IEEE.

Mallat, S. G. (1989). A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11:674–693.

Morimoto, C. H. and Mimica, M. R. M. (2005). Eye gaze tracking techniques for interactive applications. *Computer Vision and Image Understanding*, 98:4–24.

Turkan, M., Dulek, B., Onaran, I., and Cetin, A. E. (2006). Human face detection in video using edge projections. In *Proc. Int. Society for Optical Engineering: Visual Information Processing XV*, volume 6246. SPIE.

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Proc. Computer Society Conf. on Computer Vision and Pattern Recognition*, volume I, pages 511–518. IEEE.

Wu, J. and Zhou, Z. H. (2003). Efficient face candidates selector for face detection. *Pattern Recognition*, 36(5):1175–1186.

Wu, T. F., Lin, C. J., and Weng, R. C. (2004). Probability estimates for multi-class classification by pairwise coupling. *The Journal of Machine Learning Research*, 5:975–1005.

Zhou, Z. H. and Geng, X. (2004). Projection functions for eye detection. *Pattern Recognition*, 37(5):1049–1056.

Zhu, Y., Schwartz, S., and Orchard, M. (2000). Fast face detection using subspace discriminant wavelet features. In *Proc. Conf. on Computer Vision and Pattern Recognition*, volume I, pages 636–642. IEEE.

Zhu, Z. and Ji, Q. (2005). Robust real-time eye detection and tracking under variable lighting conditions and various face orientations. *Computer Vision and Image Understanding*, 98:124–154.