Robust and Accurate Eye Contour Extraction

Vladimir Vezhnevets * Anna Degtiareva

Graphics and Media Laboratory †
Faculty of Computational Mathematics and Cybernetics
Moscow State University,
Moscow, Russia.

Abstract

This paper describes a novel algorithm for exact eye contour detection in frontal face image. The exact eye shape is a useful piece of input information for applications like facial expression recognition, feature-based face recognition and face modelling. In contrast to well-known eye-segmentation methods, we do not rely on deformable models or image luminance gradient (edge) map. The eye windows (rough eye regions) are assumed to be known. The detection algorithm works in several steps. First, iris center and radius is estimated, then, exact upper eyelid contour is detected by searching for luminance valley points. Finally, lower eyelid is estimated from the eye corners coordinates and iris. The proposed technique has been tested on images of about fifty individuals taken under different lighting conditions with different cameras. It proved to be sufficiently robust and accurate for wide variety of images.

Keywords: facial features detection, eye contour extraction

1 Introduction

Automatic facial features extraction from frontal image has a wide range of usage, such as automated face modelling, facial expression recognition, face animation, feature-based face recognition. Exact eye contour conveys valuable information for all these applications.

Deformable contour models attracted by high values of image spatial luminance gradient is one well-known method for eye contour detection [Yuille et al. 1992], [Kampmann and Zhang 1998], [Yin and Basu 1999], [Lam and Yan 1996]. However, both luminance gradient (or ”image edges”) and deformable models may not be the optimal tools for eye contour extraction.

Using luminance edges as the feature for the contour detection is risky, because eye areas may have many unwanted spurious edges and lack the informative ones. This problem forces the researchers to support edge-based detection with additional features like difference between average luminance of the eye white and iris [Kampmann and Zhang 1998] or difference between saturation values of eye and surrounding skin [Yin and Basu 1999].

Moreover, deformable models, being a powerful image analysis tool need careful formulation of the energy term and close model initialization, otherwise, unexpected contour extraction result can be acquired. Frequently, a step-by-step coarse to fine shape estimation is used to lessen the number of model degrees of freedom, because active contour models tend to be attracted by local minima [Kampmann and Zhang 1998], [Yin and Basu 1999] and move away from the real contour. Sometimes many internal constraints on interdependence between the different eye features is introduced [Yuille et al. 1992] for achieving stable detection result.

Understanding the limitations of deformable models, several researchers have quit the attempts to extract complete continuous eye contour with their help and focus on several landmark points (eye corners, iris border points), from which the approximate eyelid contours are estimated [Feng and Yuen 2001], [Tian et al. 2000]. This results in less accurate, but more stable detection.

We propose a novel robust technique, which combines stability and accuracy. We suppose that rough eye regions are known. First, iris center and radius is detected by looking for a circle separating dark iris and bright sclera (the eye white). Then, upper eyelid points are found using on the observation that eye border pixels are significantly darker than surrounding skin and sclera. The detected eye boundary points are filtered to remove outliers and a polynomial curve is fitted to the remaining boundary points. In order to reduce the effect of image noise, the image is preprocessed with median filtering and 1D horizontal low-pass filtering. Finally, lower lid is estimated from the known iris and eye corners.

2 Proposed method

The input data for the algorithm is a color image containing a single human eye. The approximate scale and bounding box of the eye is considered to be known from the previous feature detection stages. The image is cropped to contain eye bounding box only and scaled to a fixed size. This normalization renders our detection techniques scale-independent and permits us to work with absolute values of some thresholds and parameters.

Our eye contour model consists of upper lid curve (in cubic polynomial), lower lid curve (in quadratic polynomial) and the iris circle. The detection is performed in four steps:

- Iris center detection
- Iris radius estimation
- Upper eyelid contour detection
- Lower eyelid estimation

The iris center and radius detection is performed in image's red-channel, which emphasizes the iris border. This is due to the fact, that iris usually exhibits low values of red (both for dark and light eyes), while the surrounding pixels (sclera and skin) have significantly higher red values. The following section describe the algorithm in detail.

2.1 Iris detection

Iris is detected in two steps - first approximate pupil center is found, then iris radius is estimated. During the radius estimation the center point is also refined.

2.1.1 Approximate iris center estimation

First, approximate iris center point is detected. The central part of the eye image is checked for strong highlight presence by comparing maximum pixel value (remember, we work with the red chan-

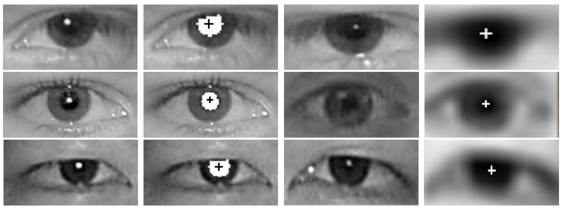
*e-mail: vvp@graphicon.ru
†www: http://graphics.cs.msu.ru

nel) to an empirical threshold. If strong highlight is found, the central area of eye bounding box is scanned with a circular search window with radius close to expected pupil (not iris) radius, checking for several conditions:

1. The local minimum inside the search window should not differ more than a given threshold from the global minimum inside the eye bounding box (this makes sure, that dark pixels are present in the search window);

2. The variance of pixel values inside the search window should not be smaller than a certain portion of global eye image variance (making sure, that both dark and bright pixels are present inside the search window);

3. The number of pixels darker than a given threshold should be not less than a pre-defined value (checking, that enough dark pixels are present inside the search window);



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

Figure 1: Approximate pupil center detection. (a, c) - red channel of eye images. (b) - the "highlight locations" and averaged result marked with cross for strong highlight case. (d) - the image after min filtering and convolution with marked mass center of darkest pixels for the case of weak highlight

All the locations, where these conditions are satisfied are called the "highlight locations" and their coordinates are averaged to determine the expected eye center (x_0, y_0). Figure 1(b) shows the typical detection result. If no strong highlight is detected, a 5x5 minimum filter is applied to eye area, to eliminate weak highlight. Then, eye image is convolved with a function, emphasizing the circular regions of dark pixels:

$$W(x,y,c) = \frac{sin((x^2+y^2)/c)}{(x^2+y^2)/c} \tag{1}$$

This function attempts weight mostly circular area at the center of convolution, the rest part of the processed block is weighted with negative values. It was introduced first in [Tsekeridou98], however, it was used for completely different purposes. We found it perfectly matching our needs. Parameter c controls the radius of region of maximum weight, it is chosen according to the expected iris size. After the convolution, mean position of 5% of darkest pixels gives the approximate iris center (x_0, y_0) - see 1(d).

2.1.2 Accurate iris detection

After the approximate iris center is known, it is refined together with iris radius detection. This is accomplished by the slightly modified algorithm developed by [Ahlberg 1999]. It makes two assumptions on the expected eye appearance: the iris is approximately circular and it is dark against the background. The iris center and radius are found by searching for a circle, which lies on the border between dark pixels of iris and bright pixel of the eye white.

To estimate the exact iris parameters (x_c, y_c, r) - center and radius, a function is defined [Ahlberg 1999]:

$$f_\Theta(x_c,y_c,r) = \int_{\theta\in\Theta} I(x_c + r\cos\theta, y_c + r\sin\theta)d\theta \tag{2}$$

where $I(x,y)$ is a graylevel eye image and $\Theta = [0, 2\pi]$. For a given (x_c, y_c) the most likely iris radius will be the value with large $\frac{d}{dr} f_\Theta$.

We apply this method to the red channel image instead of grey scale (for reasons already mentioned above) and also restrict $\Theta = [-\pi/4, \pi/6] \bigcup [5\pi/6, 5\pi/4]$, because the upper and lower iris parts are often covered with eyelids, and therefore not visible. We search through neighborhood of previously estimated iris center (x_0, y_0) and reasonable range of iris radius r, looking for high $\frac{d}{dr} f_\Theta$ values. This two-steps scheme shows high robustness and accuracy for wide class of images (see Figure 2).

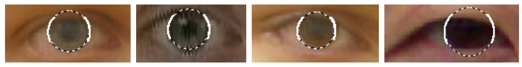


Figure 2: Iris detection results. Upper row - source images. Lower row - detected iris. The marked circle sides are used participating in iris radius detection

2.2 Eyelid detection

Our eye contour model consists of an upper lid curve (in cubic polynomial), lower lid curve (in quadratic polynomial) and the iris circle. The iris center and radius are estimated at the previous step. The upper lid detection is performed in three stages. First, a set of points that belong to upper eyelid is found. Then, this point set is examined to remove outliers. Finally, a cubic polynomial curve is fitted to the correct eyelid points. The lower lid is estimated by fitting a quadratic curve to the eye corners and the lower point of the iris circle - a reasonable approximation of the eye contour.

The well-known methods of eye shape estimation are based on using high spatial luminance gradient locations (so-called edges) as the attractors for the eyelid curve [Yuille et al. 1992], [Kampmann and Zhang 1998], [Yin and Basu 1999]. Our experiments with edge-based contour estimation techniques has shown very low reliability of this method in general case. One problem is very noisy edge map and therefore huge amount of spurious edges, even for a clear eye image. The second is possible absence or discontinuity of the significant edges, corresponding to the eyelids - in some cases the brightness transition from sclera to eye border and further to skin is too smooth to be identified as an edge by a conventional edge detector. See figure 3 for an example of an edge map suffering from both problems. These two described issues make impossible the direct usage of the edge map for eye shape estimation. Some versatile edge analysis technique, or a sophisticated edge-detector is needed in order to achieve stable and accurate results.

(a) (b)

Figure 3: An example of an eye image (a) and its spatial luminance gradient magnitude map (b). Darker pixels correspond to higher gradient magnitude

Examining the luminance values change along a single horizontal row of the eye image shows that significant local minima correspond to eye boundary points (see Figure 4). We deduce that looking for brightness valley points instead of edge points is a more appropriate way for eye shape estimation. The similar conclusion was also made by [Radeva and Marti 1995], however they used deformable models for eye contour detection. The deformable contour models object detection method is an elegant framework for reformulating image pattern recognition tasks as optimization problems. The model energy term, playing the role of the optimization criterion function, along with the model deformation rules and energy optimization strategy completely define the model behavior, forming a highly autonomous system, hardly controllable ”from without”. A subtle and carefully-tuned energy term formulation is needed for robust and repeatable detection result, which is not an easy task to come up with. Deformable models also tend to be attracted to local energy minimums, that can deviate the model from the real target contour. Close model initialization is also necessary for good detection results. We propose a novel robust technique, that avoids these difficulties, is very simple in implementation and produces accurate and reliable detection results in wide variety of images.

We detect the luminance valley points, that most likely correspond to the eye border in each row of the eye image. These points are the points of significant local minima of the horizontal luminance profiles(see Figure 4).

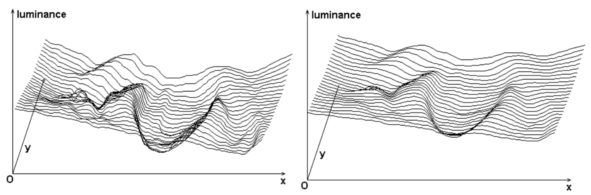


Figure 4: Pseudo-3D luminance graphs of the same eye image that produced edge map in Figure 3. Left graph - raw image data. Right graph - preprocessed image data.

Image noise can produce disturbance for luminance profiles analysis, so we added image pre-processing step for noise alleviation. First, the image is subjected to median filtering which lowers the noise level, while preserving the region borders. Then, horizontal luminance profile functions are subjected to low-pass filtering to eliminate small fluctuations (see lower graph of Figure 4 for filtering result). The low-pass filter is efficiently implemented by convolution of the eye image with 1D horizontal Gaussian mask.

The eyelid detection algorithm includes four steps: eye opening height determination, eye border points detection, outlier points removal and polynomial curves fitting.

2.2.1 Eye opening height detection

The eye opening height is determined by scanning the image iris area vertically from top to bottom, calculating each line's average red value:

$$h(y) = \frac{1}{|I_y|} \sum_{x \in I_y} R(x,y) \tag{3}$$

Here, I_y is the set of x-coordinates from the y line, that lie inside the iris circle, $|I_y|$ is the number of elements in the I_y set and $R(x,y)$ is the red value of image pixel with (x,y) coordinates. The area of low $h(y)$ values indicate the area of visible (not occluded by eyelids) iris area - the eye opening height. Figure 5 illustrates this process

2.2.2 Eye border points set construction

After the eye opening height is known, the lines of visible iris area are scanned outwards from the iris borders in search for points that satisfy one of the two conditions:

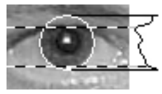


Figure 5: Detection of the eye opening height with row-by-row analysis of average iris luminance

1. the point is a start of sharp luminance increase (we have reached skin)

2. the point is a local luminance minimum, with value lower than this line's brightest sclera point minus a certain value.

This produces a set of points, that presumably lie on the upper eyelid (see figure 6). To avoid false detection of iris borders, which can occur in case of slightly inaccurate iris radius estimation, the search for eye border points along each line starts after the scanned pixels luminance starts to decrease.

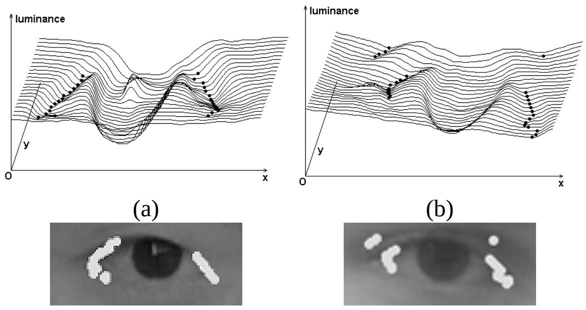


Figure 6: (a, b) - pseudo-3D luminance graphs of two eye images with eye border points set marked by dark circles. (c, d) - grayscale eye images with marked eye border points.

2.2.3 Erroneous and outlier points removal

The border points set can contain outliers and erroneous points, that would deviate the least squares fitted curve from the real eye contour. Figure 6(d) shows an example of this case. To eliminate these outliers, two straight lines are fitted to the left and right halves of the points set by the means of the hough transform [Duda and Hart 1972]. The hough transform is known to be robust to imperfect data and noise. It works with a parametric representation of the object to be detected and makes use of an accumulator space, where each dimension represents a parameter of this parametric representation. In the case of the straight line, represented by (4) there are two parameters ρ and θ.

$$\rho = x\cos(\theta) + y\cos(\theta) \tag{4}$$

We restrict these parameters to reasonable ranges for the right and left (with respect to eye the center) subsets of eye boundary points. Hough transform produces a set of lines, that pass through (or are close to, minding the discrete nature of accumulator) at least 30% of the points in the subset. The line with maximum number of point lying closer than a predefined distance ε is chosen as the *principle line* of the subset.

The points that lie too far from the principal lines are removed from the boundary points set - see Figure 7(b).

(a) (b) (c)

Figure 7: (a) - initial eye border points set; (b) - set with marked principle lines and outliers removed; (c) - a cubic polynomial curve fitted to the final border points set.

2.2.4 Eyelid curves fitting

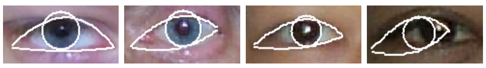
Among the remaining points the leftmost and rightmost are chosen to be the eye corners. All the points that lie above the line, connecting the eye corners are treated as belonging to upper eyelid. Finally, the upper iris border points are added to the set and the lid curve is estimated by the least squares fitting procedure. The final border point set and fitting result is shown in Figure 7(c). The lower lid is detected by fitting the eye corners and the lower point of the iris circle with a quadratic curve. The final eye contour detection results are shown in Figure 8.

3 Experimental results

The algorithm was applied to images of approximately 50 individuals taken under different lighting conditions with different cameras and quality. The typical detection results are shown in figure 8.

Figure 8: Typical detection results

Note the method robustness and accuracy for images of very different quality and lighting. Of course, the method is not perfect and sometimes detection errors occur - several examples are depicted in Figure 9.



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

Figure 9: Images where with erroneous or low-accuracy detection

One problem is the tendency to spread the eye contour too far - the result is a ”slipped” left eye corner - Figure 9(a-c). The most serious detection error due to iris detection failure is depicted in 9(d). These errors can occur when the image resolution is too small, or image sharpness and contrast are relatively low in the eye area.

4 Conclusion

The described algorithm is a part of feature detection module of an automated face modelling system. The eye shape detection is performed in three steps: approximate eye center detection, exact iris shape estimation, and eyelid curves detection. The eye center is found differently for images with frontal illumination (exhibiting strong highlight in the eye center) and ambient illumination. The iris center coordinates are refined during exact iris radius detection. The upper eyelid is found by looking for luminance valley points. The outlier points, detected falsely are rejected by utilizing hough transform. The lower eyelid is fitted to detected eye corners and the lower point of the iris. The methods we propose have shown good robustness and sufficient accuracy for face modelling application, while being simple in implementation and fast in processing time (especially compared with deformable models-based methods).

We plan to further improve the detection robustness by employing robust curve fitting techniques like RANSAC [Fischler and Bolle 1981] and its modifications or m-estimators [Stewart 1999]. This can merge the outlier rejection and eye curve fitting steps together and simultaneously increase the method robustness. Also, we plan to add lower lid detection procedure, that will help to solve the problem of ”slipped” eye corner - Figure 9(a-c). And, of course, further optimization, tuning and enhancement of the developed approaches is included in the future work by default.

5 Acknowledgements

This research was done as a part of a joint Face Modelling research project held by Graphics and Media Laboratory of Computer Science department of Moscow State University (MSU) and Multimedia Lab. of Samsung Advanced Institute of Technology (SAIT). We extend our gratitude to project heads from the MSU and SAIT side - Prof. Yuri Bayakovski and Dr. In Kyu Park.

References

AHLBERG, J. 1999. A system for face localization and facial feature extraction. Tech. Rep. LiTH-ISY-R-2172, Linkoping University.

DUDA, R., AND HART, P. 1972. Use of the hough transform to detect lines and curves in pictures. *Communication of the Association of Computer Machinery 15*, 1, 11–15.

FENG, G. C., AND YUEN, P. C. 2001. Multi-cues eye detection on gray intensity image. *Pattern Recognition 34*, 5, 1033–1046.

FISCHLER, M. A., AND BOLLE, R. C. 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communication of the Association of Computer Machinery 24*, 5, 381–395.

KAMPMANN, M., AND ZHANG, L. 1998. Estimation of eye, eyebrow and nose features in videophone sequences. In *International Workshop on Very Low Bitrate Video Coding (VLBV 98), Urbana, USA*, 101–104.

LAM, K.-M., AND YAN, H. 1996. Locating and extracting the covered eye in human face images. *Pattern Recognition 29*, 5, 771–779.

RADEVA, P., AND MARTI, E. 1995. Facial features segmentation by model-based snakes. In *International Conference on Computing Analysis and Image Processing, Prague*.

STEWART, C. V. 1999. Robust parameter estimation in computer vision. *SIAM Review 41*, 3, 513–537.

TIAN, Y., KANADE, T., AND COHN, J. 2000. Dual-state parametric eye tracking. In *Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition (FG00)*, 110–115.

TSEKERIDOU, S., AND PITAS, I. 1998. Facial feature extraction in frontal views using biometric analogiesg. In *Proceedings of the IX European Signal Processing Conference*, vol. 1, 315–318.

YIN, L., AND BASU, A. 1999. Realistic animation using extended adaptive mesh for model based coding. In *Proceedings of Energy Minimization methods in Computer Vision and Pattern Recognition*, 269–284.

YUILLE, A., HALLINAN, P., AND COHEN, D. 1992. Feature extraction from faces using deformable templates. *International Journal of ComputerVision 8*, 2, 99–111.