Eye Tracking Using Markov Models

A. M. Bagci, R. Ansari, A. Khokhar,
University of Illinois at Chicago
851 S. Morgan St. Chicago, IL
{abagci,ashfaq,ansari}@ece.uic.edu

E. Cetin
Electrical and Electronic Engineering,
Bilkent University, Ankara, Turkey
cetin@bilkent.edu.tr

Abstract

We propose an eye detection and tracking method based on color and geometrical features of the human face using a monocular camera. In this method a decision is made on whether the eyes are closed or not and, using a Markov chain framework to model temporal evolution, the subject's gaze is determined. The method can successfully track facial features even while the head assumes various poses, so long as the nostrils are visible to the camera. We compare our method with recently proposed techniques and results show that it provides more accurate tracking and robustness to variations in view of the face. A procedure for detecting tracking errors is employed to recover the loss of feature points in case of occlusion or very fast head movement. The method may be used in monitoring a driver's alertness and detecting drowsiness, and also in applications requiring non-contact human computer interaction.

1 Introduction

Driver drowsiness is among the most important causes of truck crashes. Detection of drowsiness and providing feedback to the driver about his/her alertness may reduce the risk of accidents using eye tracking. A variety of eye trackers based on image processing have been described in the literature. Deng et al. [4] presented a region-based deformable template method for locating the eye and extracting eye features. A system based on a dual state model for tracking eye features is proposed in [13]. Both of these approaches require manual initialization of eye location. A blink detection algorithm for human computer interaction has been proposed by Morris [7], in which the initialization step requires motion detection to locate the eyelids. Any other significant movement on the face, such as that induced by speaking, may cause the system to fail in detecting eyelids. De la Torre et al. [3] describes a similar system for driver warning based on principal component analysis(PCA) and active appearance models (AAM), which requires a training period for each user. Although this method is robust to translation and rotation, it may fail if the face view is not fully frontal. Recently a system based on feature detection using one camera has been suggested by Smith [11]. Eye and gaze tracking based on pupil detection using infrared (IR) illumination with special hardware have been proposed by several authors [6], [10]. These methods are robust in indoor lighting conditions while the performance may be degraded in direct sunlight.

In this paper, we propose a new method for initialization and tracking in order to overcome some of the shortcomings of earlier methods. We describe a novel initialization algorithm for facial features based on face geometry and skin color that is robust to scaling and translation. Four feature points associated with nostrils and eyebrows are detected and tracked in order to locate the eyes. After initialization an intensity-based tracking algorithm is used to track the nostrils and locate other features. Temporal evolution is captured using a Markov chain framework to model eye movement, where the states in our model correspond to an observable event, such as looking up, down etc. In our method, the face as viewed by the camera is not assumed to be frontal, which provides more accurate tracking as compared with PCA and AAM based methods. The tracking operation is robust to translation, scaling, head tilting and slight lighting changes provided the nostrils are visible. Our method may also be employed in other computer vision applications such as non-contact human computer interaction. In subsequent sections we describe the initialization and tracking algorithms, the classification procedure, and experimental results.

2 Initialization and tracking

The proposed eye tracking system uses color and geometrical features of a human face as cues to decide the direction in which a subject is looking and whether or not the subject's eyes are closed. At initialization it is assumed

Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04)
1051-4651/04 $ 20.00 IEEE

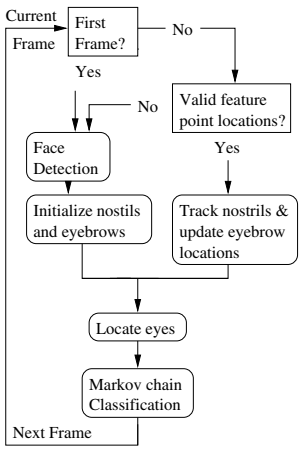

Figure 1. Flowchart of eye-tracking system

that the subject is facing the camera, with gaze directed forward. This assumption is important for adjusting the parameters for different subjects. To initialize the system, we use a skin color detector, which is robust to color variations due to different skin types. The largest blob detected with skin color analysis is assumed to correspond to the subject's face. Using geometric properties of facial features, the subject's nostrils and eyebrows are located. In video frames that follow initialization, nostrils are tracked using a procedure in which a perspective image transform is computed, following which the region containing the eye is identified and the location of the iris is extracted. This data is supplied to a Markov model module for classification. A flowchart of the algorithm is shown in Fig. 1.

2.1 Skin color detection

The distinctive color of human skin provides significant information for extracting a human face in color images.It has been shown in [8] that removing the brightness information provides a reliable method for detecting skin regions in color images. Analyzing the skin color in chromatic color space, one observes that it is confined to a small region in color space. In order to identify the cluster corresponding to the color of the skin, the pixel RGB value is mapped to normalized chromatic color space (r, g), where $r = R/(R + G + B)$ and $g = G/(R + G + B)$. The boundary of this cluster is experimentally calculated using subjects with different tones of skin color. A binary mask is computed for skin regions in the image. A connected component analysis is performed on the mask and smaller areas containing skin color are eliminated. Note that pixels corresponding to eyes, eyebrows, lips etc. do not exhibit skin color, but if they are surrounded by skin color they are also included in the mask.

Figure 2. Face mask computed with skin color detection and tracked facial features

2.2 Detection and tracking of feature points

In our approach the face is treated as a 3D rigid object, so that its motion may be defined with eight parameters or four points on the image. By tracking the location of four feature points in the video sequence we can overcome the effect of head motion such as translation, rotation and scaling. Based on the study of various facial features, nostrils and eyebrows were selected for estimating the gaze. We propose an approach based on the geometric relations of these features for detection.

When considered in triplets, eyes and nostrils form two pairs of similar triangles, the inner angles of which are known approximately. Although these angles are slightly different for each person and head pose; they can be used to detect valid combinations of these features. Candidate regions for eyes and nostrils are obtained using binary thresholding, using the fact that they contain dark pixels. The center of mass of each region is then computed. The angles subtended by each combination of three points is compared with predefined angles. The combination with minimum error is chosen, provided the error is within allowed bounds. If a good combination cannot be obtained, the bounds in the previous step are relaxed, so that more candidate regions are revealed.

Inner edges of the eyebrows are detected using the location of the nostrils. A candidate rectangular region (See Fig. 3a) containing the inner edge of each eyebrow is placed above the nostrils, at a distance which is 2.5 times the distance between nostrils. The ratio is refined for each subject after proper detection of eyebrows, with a more accurate estimate. We use a two-level Lloyd-Max quantization algorithm [5] on the gray level image to distinguish between skin and eyebrow patches in the candidate regions.

In tracking, the system normally operates in the simpler of two modes for locating the nostrils and the eyebrows. The region around these features is composed of skin, so a simple thresholding operation reveals the correct locations. This method provides robustness to illumination changes, as the nostrils are considerably darker than the surround-

Figure 3. Search regions for nostrils and brows, and relative locations of features

ings in most illumination conditions. The search regions are indicated in Fig. 3a.

2.3 Perspective transform

The next step in the algorithm is finding the location of the iris. An initial estimate for a bounding box containing the eyes is calculated in the first frame. When the location of the iris is confirmed after processing the first frame, the bounding box is centered on that point. In the following frames, the bounding box is projected on the face image, and the region containing the eye is determined. The projection used for the perspective transform is described in Equation 1.

$$\begin{bmatrix} u' \\ v' \\ w \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (1)$$

where (x, y) are pixel coordinates in the initial frame. The corresponding coordinates (u, v) in the projected frame are $u = u'/w$ and $v = v'/w$.

The elements of the transformation matrix can be derived using the following set of equations,

$$u_i = a_{11}x_i + a_{12}y_i + a_{13} - a_{31}x_iu_i - a_{32}y_iu_i \quad (2)$$

$$v_i = a_{21}x_i + a_{22}y_i + a_{23} - a_{31}x_iv_i - a_{32}y_iv_i \quad (3)$$

where (x_i, y_i) and (u_i, v_i) are corresponding pixel coordinates in initial and projected frames respectively. For a complete description of the system, four points are needed in each frame, which are the nostril and the eyebrow coordinates in our case. The center of mass of iris identified after the bounding box is calculated. The light-colored regions such as the skin or the sclera are filtered out using the Lloyd-Max quantization algorithm with an adaptive threshold, followed by a morphological opening operation with a circular structuring element. Whether or not the eyes are closed is decided at this step If the eyes are closed the morphological filter removes the falsely detected regions such as eyelids.

2.4 Detection of tracking errors

The system may lose tracking due to occlusion or very fast head movement. Tracking errors are detected using a geometrical face model. As seen in Fig. 3b eyebrow edges and nostrils form a trapezoid, inner angles of which are determined in the first frame. The inner angles of the trapezoid are robust to scaling and translation of the face. In case of rotation of the head to either side skews the trapezoid but the inner angles still stay within certain limit. If the geometry is considerably different, the system should be initialized in order to run the skin detection algorithm.

3 Classification using Markov Models

Markov models are stochastic models used in analysis of time varying signals. Hidden Markov models have been used widely in speech [9], pattern and motion recognition [12]. Several variations of HMMs have also been proposed for computer vision such as coupled or multidimensional models [1]. Here we use the simpler form of Markov models to model eye movement, as each state in our model corresponds to an observable event, such as looking up, down etc.

We train the model using Baum-Welch algorithm [9], with observation vectors of normalized (x, y) iris coordinates, which were aligned with the training data using the perspective transform in Equation 1. Gaze is quantized to five states corresponding to position of the iris (such as looking up, down, left, right and forward). After classification, the model parameters, $\lambda = (A, B, \Pi)$, are obtained, where elements of $A = [a_{ij}]$, denote the state transition probabilities between states i and j, $B = [b_j(O)]$ the probability of making the observation O at state i and π_i denotes the probability of system being in state i initially. These parameters along with the feature point locations are stored for classification step.

The observation data we obtained from the tracking module may be classified immediately for instant feedback to the user depending on the application. The optimal state sequence associated with the given observation sequence is determined using the Viterbi algorithm [9]. The label (such as "left", "up" etc.) corresponding to the optimal state is displayed as classification result.

4 Experimental results

Performance of the system is evaluated using a set of video clips consisting of a total of 13000 frames of five different subjects of different ethnicities. The videos were shot in indoors, with slightly varying illumination conditions, using a Canon GL2 camera. The frame size is 720x480

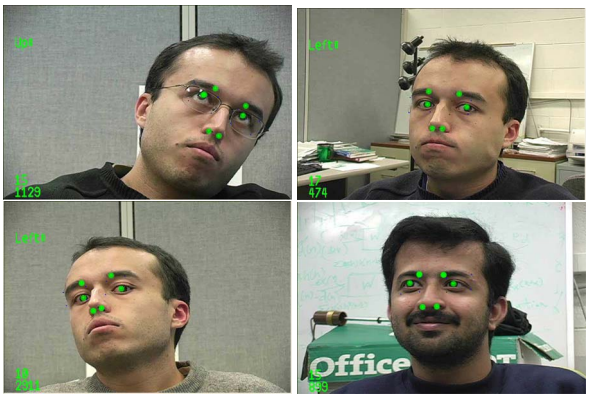

Figure 4. Samples of output images

pixels, and face occupies approximately 20% of the image size. The tracker successfully locates the features in 99.2% of the frames. It does not lose track unless there is a drastic lighting change, fast head movement, or occlusion, provided both the nostrils are visible. Closed eyes are detected with an error rate of 1.2% The classifier module determines the location of the iris in 98.5% of the frames, excluding closed eyes. The detection and classification errors usually occur in extreme head poses, for full-frontal faces the classification rate is close to 100%. One example where the system fails is shown in Fig. 5. The subject is turning his head to his left in these consecutive frames (at 30 fps), and the tracking fails as the rotation continues. The tracking resumes once both nostrils are visible again. The correct classification rates (95%) for subjects wearing glasses are lower because frames and reflections on the glass may occlude the iris. Tracking is not affected provided eyebrows are not occluded by frames. Sample frames from output videos are shown in Fig. 4. We also implemented a tracker based on active appearance models [2],[3] to compare the performance with our algorithm. To obtain the learning matrix, 1650 synthetically perturbed images are used. The perturbations allowed are 4 pixels for translation, 20% in scale and rotation changes of $\pm 15^{\circ}$. AAM based method maintains accurate tracking in 93% of the test data, as compared with 99.7% accuracy of our method in the same dataset. Tracking errors in AAM based method usually occur when the face view is not fully frontal or when the head is tilted more than 20° to the side.

5 Conclusion

An eye detection and tracking algorithm based on color and geometrical features of the human face has been presented. The system possesses automatic facial feature initialization and tracking error detection and it is robust to

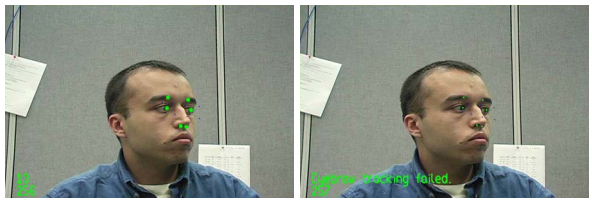

Figure 5. Head rotation where tracking fails.

lighting variances in indoor conditions, scaling, translation and different orientations of head.

To use the system in a car environment, the system should be improved to tackle severe lighting changes. We are looking into methods for improving computational efficiency, as the processing rate is around 3 fps on a Pentium 4 1.7GHz processor using frames of size 720x480 pixels. Future work will focus on using multiple cameras and deciding the head pose, which is also an indicator of drowsiness.

References

[1] M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. *Proc. of CVPR*, pages 994–999, 1997.

[2] T. F. Cootes, G. J. Edwards, and C. J.Taylor. Active appearance models. *Proc. ECCV*, 2:484–498, 1998.

[3] F. De la Torre, C. Garcia Rubio, and E. Martinez. Subspace eyetracking for driver warning. *Proc. of ICIP*, 3:329–332, 2003.

[4] J. Deng and F. Lai. Region-based template deformation and masking for eye-feature extraction and description. *Pattern Recognition*, 30(3):403–419, 1997.

[5] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice Hall, 1988.

[6] Q. Ji. Face pose estimation and tracking from a monocular camera. *Image and Vision Computing*, 20(7):499–511, 2002.

[7] T. Morris, F. Zaidi, and P. Blenkhorn. Blink detection for real-time eye tracking. *J. Networking and Computer Applications*, 25(2):129–143, 2002.

[8] N. Oliver, A. Pentland, and F. Berard. Lafter: Lips and face real time tracker. *Proc. of CVPR*, pages 123–129, 1997.

[9] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):257–286, Feb 1989.

[10] R. Ruddarraju, A. Haro, and I. Essa. Fast multiple camera-head pose tracking. *In Proceedings VI*, 2003.

[11] P. Smith, M. Shah, and N. da Vitoria Lobo. Determining driver visual attention with one camera. *Trans. on Intelligent Transportation Systems*, 4(4):205–218, 2003.

[12] A. Sundaresan, A. RoyChowdhury, and R. Chellappa. A hidden markov model based framework for recognition of humans from gait sequences. *Proc. of ICIP*, 2:93–96, 2003.

[13] Y. Tian, T. Kanade, and J. Cohn. Dual-state parametric eye tracking. *Proc. of Conf. on Automatic Face and Gesture Recognition*, pages 110–115, 2000.