Tracking Eyes and Monitoring Eye Gaze

Rainer Stiefelhagen, Jie Yang, Alex Waibel
stiefel@ira.uka.de, yang+@cs.cmu.edu, ahw@cs.cmu.edu

Interactive Systems Laboratories
University of Karlsruhe — Germany, Carnegie Mellon University — USA

ABSTRACT

In this paper, we present a non-intrusive eye tracker which can detect and track a user's eyes in real time as soon as the face appears in the view of the camera without special lighting or any marks on the user's face. We also discuss the problem of gaze monitoring. We employ neural networks to estimate a user's eye gaze using the images of user's both eyes as input to the neural networks. We have collected 4 sets of data from 4 different users using the eye tracker and have trained and tested several neural networks. The eye gaze monitoring system has achieved accuracy of between 1.3 and 1.8 degrees with a user dependent setup and of 1.9 degrees for a multi-user setup.

1. INTRODUCTION

Focus of attention plays an important role in user modeling. In many applications, it required to monitor user's eye gaze to find user's focus of attention.

Current eye gaze tracking methods basically rely on intrusive techniques such as measuring the reflection of some light (ususally infrared light) that is shone onto the eye, measuring the electric potential of the skin around the eyes or applying special contact lenses that facilitate eye gaze tracking [2].

Baluja and Pomerleau proposed a non-intrusive method to estimate the eye-gaze based on a neural network [1]. It has been demonstrated that the neural network could accurately estimate the position of the eye gaze on a computer screen given images of the user's eyes as input. However, the system used an active sensing approach by shining light into the user's right eye. This causes the problem of user's acceptance. In order to avoid using the flash light, a high performance eye tracker is required.

In this paper, we present a non-intrusive eye tracker that can detect and track a user's eyes as soon as the face appears in the view of the camera without special lighting or any marks on the user's face. We also discuss the problem of gaze monitoring. We employ neural networks to estimate a user's eye gaze using the images of user's both eyes as input. We have collected 4 sets of data from 4 different users. We have trained and tested several neural networks. Experimental results have demonstrated feasability of the concept and system.

2. TRACKING EYES AND PREPROCESSING

2.1. Tracking the face

To locate the eyes we first try to find the face in the camera image. To find and track the face, we use a statistical color-model consisting of a two-dimensional Gaussian distribution of normalized skin colors [4]. The input image is searched for pixels with skin colors and the largest connected region of skin-colored pixels in the camera-image is considered as the region of the face. Figure 1 shows the application of the skin color model to a sample input image. The color-distribution is initialized so as to find a variety of skin-colors and is gradually adapted to the actual found face. Once the face is found in the image, the search for the eyes is restricted to the found facial area.

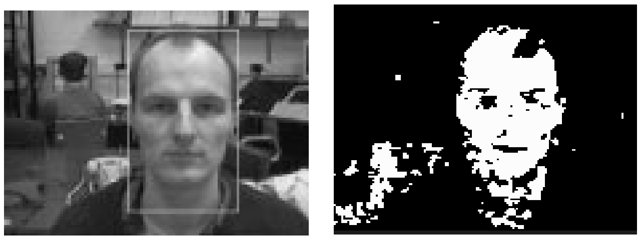
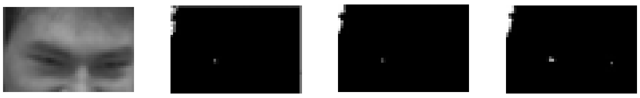


Figure 1. Locating the face using the skin color classifier

2.2. Searching and Tracking Eyes

Assuming a frontal view of the face initially, we can search the pupils by looking for two dark regions that satisfy certain anthropometric constraints and lie within a certain area of the face.

For a given situation, these dark regions can be located by applying a fixed threshold to the grayscale image. However, the threshold value may change for different people and lighting conditions. To use the thresholding method under changing lighting conditions, we developed an iterative thresholding algorithm. The algorithm iteratively thresholds the image until a pair of regions that satisfies the geometric constraints can be found. Figure 2 shows the iterative thresholding of the search window for the eyes with thresholds k_i. After three iterations, both pupils are found.



$k_0 = 30$ $k_1 = 32$ $k_2 = 34$

Figure 2. Iterative thresholding of the search window

Because the thresholding value is adjustable, this method is able to apply to various lighting conditions and to find the pupils in very differently illuminated faces robustly.

Figure 3 shows the search windows for the eyes and found eyes for two faces.

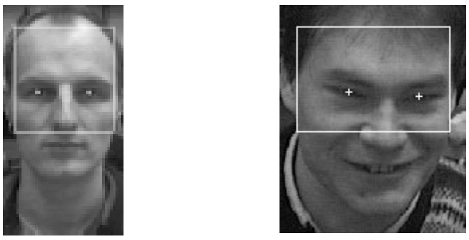


Figure 3. Search area for eyes

Once the pupils are found, they can be tracked in the following frames by simple darkest pixel finding in small search windows around their current position. The search windows furthermore can be predicted using a translational motion model.

Note tha in our actual system we are not only tracking the eyes but also lip corners and nostrils. Using all these facial points and the 3D locations of these points on a human head we then can compute the head translation and rotation (pose) and we are able to detect outliers and to recover from tracking failures. Details of the tracking of all facial features, and the methods to detect and recover from tracking failure are described in detail in [3].

Tracking of the facial features is done in real time and runs at a frame rate of 15-30 frames per second, depending on the size of the found face in the image.

2.3. Preprocessing

After the pupils are located in the camera image, a small window around each pupil in the camera image is extracted and stored to disk. The extracted images then are histogram equalized in order to compensate for different lighting conditions and used as input for the net. Figure 4 shows two sample pairs of extracted and preprocessed eye images that are used as input to the neural nets.

Figure 4. Sample input images for the neural net (20x10 pixel)

2.4. Stabilizing the input image using template matching

We discovered that using our eye tracker to extract the eyes, the eyes were sometimes shifted by a few pixels in the extracted images. For our first training experiments we just took those images for training.

In the next experiment we performed template matching with two eye templates in a small region around the tracked pupil position. We then extracted the eyes around the position which led to the best match with the templates. Using these "stabilized" eye images we could slightly improve the accuracy of the eye gaze estimation in most cases (see Table 1 and 2).

3. ESTIMATING EYE GAZE WITH A NEURAL NET

Baluja and Pomerleau demonstrated that a neural network could achieve accurate eye gaze estimation on the screen if stable images of user's eyes could be acquired [1]. Motivated by their work, we have trained several neural networks to estimate a user's eye gaze on a computer screen using the eye images obtained with our eye tracker. We use the structure of a three layer network. Our experience shows that 40 to 50 hidden units are appropriate for the task. Both eye regions with a size of 20 x 10 pixels each are fed into the neural network, leading to 400 input units. The output of the nets consisted of 2 x 50 output neurons for Gaussian output representation of the x- and the y-coordinates of the focused point on the screen. The neural nets are trained by standard backpropagation method.

3.1. Data Collection

We collected 4 sets of images from 4 different subjects. Each set contains about 5000 training and testing samples (two samples for each screen position). The setup consists of a computer and a camera mounted below the screen. Each subject was asked to sit in front of the computer and follow the cursor on the screen with his eyes. The cursor was then moved horizontally and vertically all over the screen, controlled by the recording program. The eye tracking module automatically located the eyes of the subject in the camera images and extracted both eye regions. Both the eye regions and the cursor locations were stored for training and testing of the gaze tracker.

3.2. Results

We have trained and tested both user dependent and a multi-user gaze tracker. To train user dependent gaze estimation networks we divided the data for each user randomly in a training set of 4000 samples and a test set of 1000 samples. First we trained the nets with the histogram equalized eye-images as extracted by the eye tracker without doing additional template matching. Using this approach we received accuracy between 1.4 degrees and 2.0 degrees, depending on the data set. Table 1 shows the best results for each of the persons.

We also trained nets with the eye images that were extracted using additional local template matching with eye templates (as described in section 2.4.). Using these "stabilized" images, the accuracy increased by 0.2 degrees for three out of the four sets leading to accuracy between 1.3 and 1.8 degrees (see Table 1).

input images	set1	set2	set3	set4
histo.equalized	1.4	1.5	1.8	2.0
histo.equ. + stabilized	1.5	1.3	1.6	1.8

Table 1. person dependent accuracy (in degrees)

For the training and testing of a multi-user system, we combined each of the training sets and each of the test sets. Here we achieved 2.3 degrees accuracy without stabilizing the eye images. Using stabilized input images the accuracy increased to 1.9 degrees as shown in Table 2.

input images	sets 1,2,3,4
histo.equalized	2.3
histo.equ. + stabilized	1.9

Table 2. multi person accuracy (in degrees)

4. CONCLUSION AND FURTHER RESEARCH

We have presented a non intrusive real time eye tracking and gaze monitoring system. The system is able to locate and track a user's pupils as soon as the user appears in the view of the camera without special lights or any mark on the user's face. We have employed the eye tracking system to extract the regions of the eyes for monitoring a user's eye gaze based on the trained neural network. The gaze tracking system has achieved accuracy between 1.3 degrees and 1.8 degrees for user dependent neural nets and accuracy of 1.9 degrees for a multiuser net.

One of the problems in the current gaze tracking system is that only local information, i.e., the images of the eyes, is used for estimating the user's gaze. Consequently the system relies on a relatively stable position of the users head with respect to the camera and the user should not rotate his head.

To make the gaze tracking system more robust against user movement, it would be helpful to also use additional information such as the 3D position of the head relative to camera to estimate the users gaze. We are currently working on combining user's eye gaze information with information of user's head orientation.

In the current system the problem of deriving the focus of attention from the user's 'low level' eye gaze patterns has not yet been adressed. In fact, even if we could have a perfect gaze tracking system, we still have the problem to find a user's focus of attention using only the gaze information. A high level user model is needed to deal with involuntary eye-movements. We will adress this problem in our further research.

For additional information please have a look at our web-site at

http://www.is.cs.cmu.edu/ISL.multimodal.html

ACKNOWLEDGEMENTS

This research was sponsored by the Advanced Research Projects Agency under the Department of the Navy, Naval Research Office under grant number N00014-93-1-0806.

REFERENCES

[1] Shumet Baluja and Dean Pomerleau. Non-intrusive gaze tracking using artificial neural networks. Technical Report CMU-CS-94-102, Carnegie Mellon University, 1994.

[2] Arne John Glenstrup and Theo Engell-Nielsen. Eye controlled media: Present and future state. Technical report, University of Copenhagen, http://www.diku.dk/users/panic/eyegaze/, 1995.

[3] Rainer Stiefelhagen, Jie Yang, and Alex Waibel. A model-based gaze tracking system. In *Proceedings of IEEE International Joint Symposia on Intelligence and Systems*, pages 304 – 310, 1996.

[4] Jie Yang and Alex Waibel. A real-time face tracker. In *Proceedings of WACV*, pages 142–147, 1996.