Detection and Tracking of Eyes for Gaze-camera Control

Shinjiro Kawato and Nobuji Tetsutani
ATR Media Information Science Laboratories
{skawato | tetsutani}@atr.co.jp

Abstract

We propose new algorithms to extract and track the positions of eyes in a real-time video stream. For extraction of eye positions, we detect blinks based on the differences between successive images. However eyelid regions are fairly small. We propose a method to distinguish them from head movement. For eye position tracking, we use an updating template based on a "Between-the-Eyes" pattern instead of the eyes themselves. Eyes are searched based on the current position of "Between-the-Eyes" and their geometrical relations to the position in the previous frame. The "Between-the-Eyes" pattern is easier to locate accurately than eye patterns. We implemented the system on a PC with a Pentium III 866MHz CPU. The system runs at 30 frames per second and robustly detects and tracks the eyes.

1 Introduction

Head-free and head-off gaze detection systems can provide a good interaction interface for computers. A head-off gaze-camera captures an eye image and analyzes it to estimate the gaze direction. To increase the accuracy of this estimation, the eye image must be taken at high resolution. Consequently, the view-field is fairly small, typically about 4×3 cm so that an eye is fully in the image, and the depth of the focus is very shallow. With only a gaze-camera, users cannot move their heads. To make the system head-free, some other means is required to detect and track the eye in order to control pan, tilt, and focus of the gaze-camera.

This paper proposes algorithms for eye detection and tracking that allow such a system to control a gaze-camera but not gaze detection itself.

Matsumoto has reported excellent eye tracking performance in his gaze direction measurement system[5]. However each user has to register his/her face and feature points beforehand. We also use video cameras to detect and track eyes, but we want to treat this subject more generally and only take template patterns on the fly.

The situation we assume is that a user is sitting in front of a gaze-camera at a distance of 50—100 cm and is looking at a display monitor. In this situation, the range of face movement is not large, so we can use a fixed camera to take a face image with enough resolution to detect eyelid movements or blinks. Eyelid location means eye location.

A human blinks involuntarily and periodically. It doesn't take us much time to wait for natural blinks. In the case of using gaze detection as an interface, we can even expect a user's voluntary blink. The fact that both eyes blink at the same time provides useful information for distinguishing the blinking from other motions in the scene.

In a recent survey paper on face detection[3], only one report[2] was mentioned in which the authors detected faces from blinking. Their blink detection method was based on the differences between successive images. According to their description, significant differences in luminance appear only in the small boundary region around the outside of the head and eyelid movement regions. This means the head is nearly still, at least between the two successive images when blinks occur.

In another paper[1], eyes were also located by blink detection. First, they extracted a face region based on a combination of background subtraction and skin-color information. Then they analyzed luminance differences between successive images in the face region to extract blinking. However, there was no description of head movement compensations. Therefore, the head is assumed to be nearly still.

We propose another blinking detection algorithm based on the differences between successive images, which distinguishes eyelid movements from head movement so that it detects blinks even while the head is moving. This function increases the flexibility of applications.

Once we detect locations of the eyes, we have to track them in successive images. Considering the changes in face orientation, we cannot apply a simple template matching technique. Even when we update the templates frame-by-frame with patterns of current located positions, the tracking points will gradually migrate out from the eyes, because the tracked points are not always the centers of the eye pattern.

In the report [1], each eye is tracked with a combination of a fixed template and an updating template. The fixed eye template is taken when a blink is detected, and the updating template is taken at located eye position in the latest image. When they search for an eye with these template patterns, they put 3/4 weight on the fixed template and 1/4 on the updating one. The reason for keeping the initial template is not