# Enlargement of a Movement Area and Processing Multi Persons in a Vision-Based Face Tracking System

Yasuto Nakanishi Atsushi Nagasawa Hideki Koike
Graduate School of Info. Sys., Univ. of Electro-Communications
1-5-1 Chofugaoka, Chofu, Tokyo Japan
+81424-43-5653
{naka, atsushi, koike}@vogue.is.uec.ac.jp

Yoichi Sato
Inst. of Industrial Science, Univ. of Tokyo
4-6-1 Komaba, Meguro, Tokyo Japan
+813-5452-6461
ysato@iis.u-tokyo.ac.jp

## ABSTRACT
We have developed a stereo-based face tracking system which can track the 3D position and orientation of a user in real-time, and have improved it for a large display. Our original system could track only one person, and the area in which he/she could move around was small. In this paper, we describe the enlargement of the movement area and the ability to track multiple persons in our face tracking system. Our tracking system incorporates the dynamic update of template images for tracking facial features so that the system can successfully track a user who moves around in front of the camera units; it also utilizes some image-processing boards for tracking multiple persons. These features would be necessary for ubiquitous computing environments using large-sided displays.

## Keywords
Perceptual user interface, face tracking, image-processing.

## 1. INTRODUCTION
Recently, large-sized displays such as plasma displays or LCD projectors that can project images to a large area have become popular. They are often used in public places, e.g., train stations or shopping malls for displaying information. However, a large-sized display with equipped with a touch sensor has become popular gradually, it needs the positive action of a user to do so. Large-sized displays in public spaces often show advertisements or information in the form of movies, and they would draw attention than general information kiosks. A face-tracking system would be able to give information to users who do not approach to the display actively or to those who happened to pass the display. Using the eyes or the face as a source of input in advanced user interfaces has long been a topic of interest to the human computer interaction field. With using a face-tracking system, we can also record the number of occasions when each advertisement or information has been watched. It would be possible to offer the information about which advertisement is popular among people and the information about marketing to the person who offered advertisements or information [1].

This paper was presented at "Physical Interaction (PI03) - Workshop on Real World User Interfaces", a workshop at the Mobile HCI Conference 2003 in Udine (Italy). September 8, 2003. The copyright remains with the authors.
Further information and online proceedings are available at http://www.medien.informatik.uni-muenchen.de/en/events/pi03/

We have also developed a face-tracking system that utilizes incorporates dynamic update of template images for tracking facial features so that it can successfully track a user's face for a large angle of rotation, and implemented several prototype applications [8]. However, our system tracked the face of a single user and did not work on multiple people simultaneously. We utilized some fixed parameters for finding facial features and for processing template-matching method, and the area in which our system tracked a user's face was not so large that he/she could move around in front of the system. A large display in a public space would be watched with multi persons, and they will watch it from various distance. In this paper, we describe enlargement of a movement area and tracking multi persons in our face tracking system.

There are several kinds of commercial products that detect a human's head position and orientation using magnetic sensors and link mechanisms, and there are much research based on computer vision techniques [1-4, 8, 9]. Haro presented a real-time pupil detector and tracker that utilized a probabilistic framework [3]. They used an infrared lighting camera to capture the physiological properties of eyes, Karman trackers to model eye/head dynamics, and a probabilistic-based appearance model to represent eye appearance. Kawato proposed an approach that tracks a point between the eyes and then locates the eyes [4]. It utilizes an image filter, i.e., the circle-frequency filter to detect "between-eyes," and stores the small area around it as a template for template matching. Stiefelhagen presented an eye tracker without special lights that employs neural networks to estimate a user's eye gaze using the images of both of the user's eyes as input [9]. They trained several neural networks to estimate a user's eye gaze on a computer screen using the eye images obtained with their eye tracker. However, most of these systems utilize a monocular image and it is very difficult to compute the full 3D locations and orientation of a face or to detect the eye gaze direction accurately and robustly. The most relevant work to us is by [6]; that work employs the template matching method for detecting the edges of eyes and a mouth by using a stereo camera pair. Their system tracks the 3D coordinates of the facial features and aims to utilize them as a visual human interface for a cooperative task with a robot. These studies, however, assume that a user sits down in front of a computer monitor. Our purpose in this research is to develop a face-tracking system not for a personal display, but rather for a large-sized display. Darrel et al. explored three different interface paradigms using a face-tracking system: direct manipulation, gaze-mediated agent dialog, and perceptually-driven remote presence, and showed a face-tracking system is an important module in designing perceptual interfaces for intelligent environments [2]. However, most of the previously developed face-tracking systems were designed to be used by a user sitting in front of a monitor; therefore, they might not be suitable for applications with a large-sized display as an ubiquitous computing environment.

## 2. Our Face Tracking System

### 2.1 Overview

Our vision-based face tracking system can track the position and orientation of a user in real-time (30 frames/sec) [4]. The configuration of our tracking system is similar to the one proposed by Matsumoto et al. [5], but our tracking system is capable of tracking a user's face for wider angles of rotation by introducing dynamic update of template images [8]. Our system runs on a PC (Pentium4-2.8GHz, Linux OS) equipped with a HITACHI IP5010 image-processing board, which is used while being connected to two NTSC cameras. It is equipped with 40 frame memories of 512 x 512 pixels. In order to reduce the processing time of face tracking, we use the lower resolution image whose size is 256 x 220 pixels. We use a camera unit that consists of two 3CCD black-and-white cameras and two near-infrared lights; the disparity of the two cameras is 16 cm (Figure 1). The cameras are equipped with infrared filters. These filters transmit only the light whose wavelength is close to infrared rays. By using this filter, the camera takes only the infrared light that reflects in the face of the user, thereby enabling us to eliminate the background images.

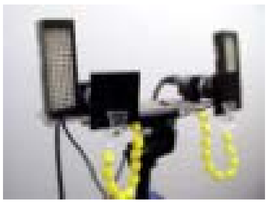

**Figure 1. The stereo camera unit in our face-tracking system.**

### 2.2 Stereo Tracking Algorithm

In order to search facial features from the camera images, we first select the region of the face. This is done by binarizing an input image from each camera while changing the threshold of binarization iteratively. Then, within this extracted facial region, we identify the location of pupils with the algorithm proposed by Stiefelhagen [9]. We search for the pupils by looking for two dark regions that satisfy the creation of anthropometric constraints and lie within a certain area of the face. After the pupils are located in the camera image, we identify the location of the mouth based on histogram projection in two orthogonal directions.

After storing the template images, we perform the template matching with four template images of eye edges and with two template images of mouth edges for each camera image. This search process using template matching is computationally expensive. Therefore, search areas are defined in our method and the eye edges and the mouth edges are searched for only within these areas instead of over an entire region of the user's face. In this process, each feature is assumed to have a small motion between the current frame and the previous one. We perform the template matching only in the areas around the eye and mouth locations that were found in the previous frame. The areas of a fixed size, e.g., 48 x 38 pixels in our current implementation, are set so that they include the locations of the edges of the eyes and the mouth obtained at the previous frame. We utilize a function of normalized correlation equipped in the image-processing board in template matching, and six 2D locations are found for each camera image. Then the 3D coordinate of each feature is determined based on triangulation.

The locations of the eye and mouth edges found in template matching are obtained independently, and the provided 3D coordinates do not always correspond to the model of the face registered at the initialization. There might be the case that multiple candidates exist for matching and that inappropriate points are detected, and it would not be appropriate that we utilize those locations. We utilize the 3D model of the face stored at the initialization to cope with this problem. We revise the coordinates provided in template matching so that they retain the nature of the rigid body model. We use the algorithm that lets a rigid body model fit the last state in the previous frame using the virtual springs proposed in [5].

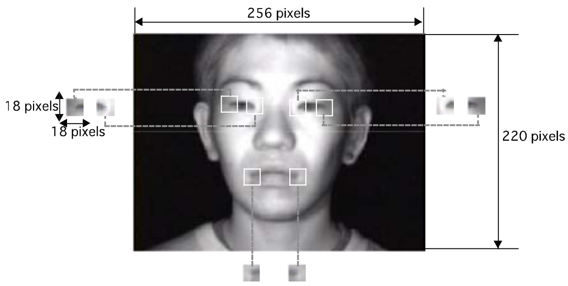


**Figure 2. Samples of obtained initial templates.**

### 2.3 Enlargement of a Movement Area

Our previous system has fixed parameters for searching facial features and for template matching. When a user move forward and back in front of our camera unit, the sizes of template images don't match with the user's face size, and our previous system fails to keep tracking.

In order to enlarge an area where a user can move, we dynamically change those parameters and store new template images according to the distance between the user and our camera unit. We set five areas, and change the parameters according to the area in which a user is (Figure 3).

Figure 4 shows the new algorithm for searching facial features. After selecting the region of the face, we estimate one area in which a user is using the face region size.

The threshold values for this estimation were decided with averaging face region sizes of five persons at each area.

According to the selected area, we select parameters for searching eye edges and mouth edges and template image size. With the selected parameters, we detect facial features and store the initial template images. This makes it possible to detect facial features even if a user enters into any area, however, we should also update template images dynamically in order to keep tracking a user when he/she moves. Figure 4 also shows the new algorithm for tracking eye edges and mouth edges.

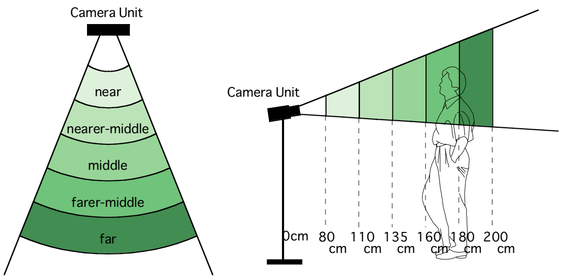


**Figure 3 The five areas in our face-tracking system.**

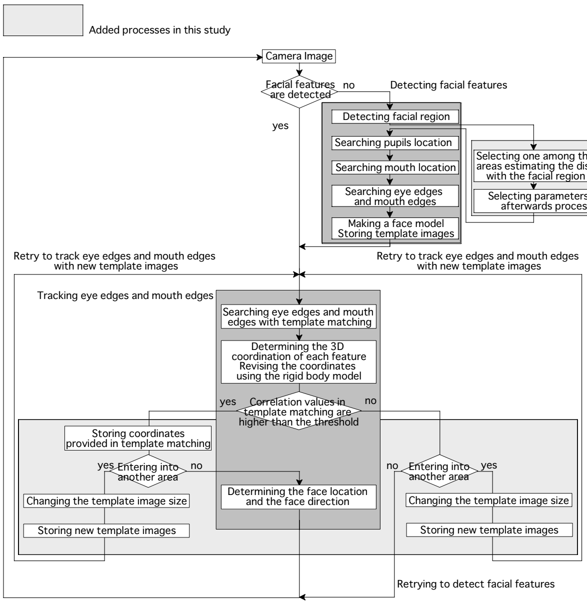


**Figure 4 The improved algorithms.**

After searching eye edges and mouth edges and obtaining the 3D coordinate of each feature, we check the obtained correlation values in template matching.

When the correlation values are higher than the predefined threshold, we store the provided 3D face location once and check whether the user enters into another area with the location. When the user remains one area, we determine where the user looks at using the face location and the face direction. When the user enters into another area, we change the template image size, store new template image at the obtained 3D coordinates and retry template matching with the new template images.

When the correlation values are lower than the predefined threshold, we check whether the user enters into another area with the obtained face location. When the user remains one area, we retry to detect facial features and restart tracking. When the user enters into another area, we change the template image size, store new template images at the obtained 3D coordinates and retry template matching with the new template images.

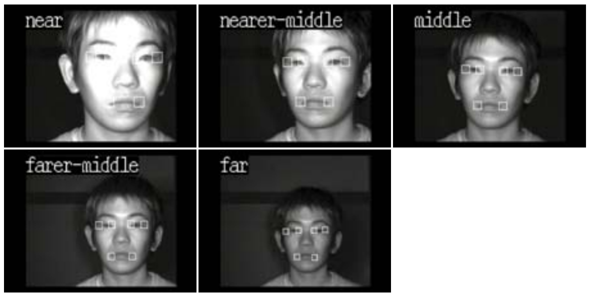


**Figure 5 Examples of the tracking result in each area.**

In our previous system, the area size that a user could move forward and back was 30 cm, which is just the "near" area among in Figure 3. In our current system, the current area size has become 120 cm, which is four times of the previous system.

## 2.4 Processing Multi Persons

Our previous system tracked only one person, and its one reason was that one user's face occupied most of the camera image as shown Figure 2. With the enlargement of the movement area, it has become possible to get an image of two person's faces with our camera unit.

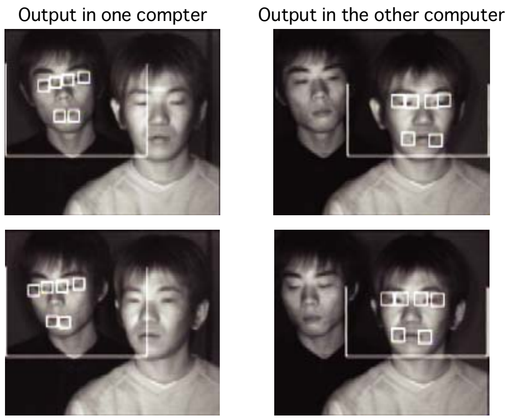


**Figure 6 Output image in each PC**

The image processing board that we utilize doesn't correspond to a multi-thread program, and the processing speed has become slow when tracking two persons with processing in one loop. For these reasons, we utilize two PCs equipped with an image-processing board in the current implementation. The camera image is distributed to the two image-processing board, and each board processes different area among the same image.

When each board processes half of an image, there was a case that one person's face was in the half of the image from one camera but was not in the half of the image from the other camera. In the current implementation, each board processes an area of two-thirds of an input image (Figure 6).

## 3. Discussions and Conclusions

In this paper, we presented a stereo-based face tracking system which can track the 3D position and orientation of a user in real-time, and the expansion for enlargement of a movement area and for processing multi persons. Our tracking system incorporates dynamic update of template images for tracking facial features so that the system can successfully track moving users' faces. Another advantage of our tracking system is that it does not require a user in the area to manually initialize tracking process, which would be critical for natural and intuitive interaction in ubiquitous computing environments. Some researches can track multi persons, however, they tracks only 2D locations and assume that users sit down in front of a computer monitor. On the other hand, our system track 3D locations and we assume that users are in front of a large display move around.

In the current implementation, we utilize the face region size for estimating one area in which a user is, and the threshold values for this estimation process are heuristic. With using other systems such that can detect pupil easily [3, 4] or such that know a position of a user with other stereo camera units jointly, our tracking system will become more robust. For tracking two persons, we utilize two PCs equipped with an image-processing board. Because this system constitution is not suited for tracking more persons' faces, we are porting our system by using a software image-processing library instead of the image-processing board [10]. With this porting, it will become easy to add more camera units and develop a multi-thread program for tracking multi persons. We will enlarge the area in which a user can move from side to side by adding some camera units and will track more persons' faces

## 4. REFERENCES


[1] Crowley C, J., Coutaz, J. and Berard, F., Things that See: Machine Perception for Human Computer Interaction, Communications of ACM, 43, 3, 54-64,2000.

[2] Darrell, T., Tollmar, K., Bentley, F., Checkac, N., Morency, L., Rahimi, A. and Oh, A., Face-responsive interfaces: from direct manipulation to perceptive presence, Proceedings of $4^{th}$ International Conference on Ubiquitous Computing (UBICOMP2002), pp. 137-151, 2002.

[3] Haro, A., Flickner, F, M., and Essa, I., Detecting and Tracking Eyes By Using Their Physiological Properties, Dynamics, and Appearance, Proceedings IEEE CS Conference on Computer Vision and Pattern Recognition (CVPR 2000), pp. 163-168, 2000.

[4] Kawato, S. and Ohya, J., Two-step Approach for Real-time Eye Tracking with a New Filtering Technique, Proceedings of IEEE International Conference on System, Man & Cybernetics, Nashville, pp. 1366-1371, 2000.

[5] Author1., Author2. and Author3, Vision-based face tracking system for window interface: prototype application and empirical studies, Extended Abstracts of 2001 ACM Symposium on Human Factors in Computing Systems (SIGCHI2001), pp. 359-360, 2001.

[6] Matsumoto Y. and Zelinsky, A., An Algorithm for Real-time Stereo Vision Implementation of Head Pose and Gaze Direction Measurement, Proceedings of IEEE International Conference on Face and Gesture Recognition (FG'2000), pp. 499-505, 2000.

[7] Morimoto, C. H., Koons, D., Amir, A. and Flickner, M., Pupil detection and tracking using multiple light sources, Technical Report RJ-10117, IBM Almaden Research Center, 1998.

[8] Author1, Author2, Author3, Author4 and Author5, Vision-Based Face Tracking System for Large Displays, Proceedings of $4^{th}$ International Conference on Ubiquitous Computing (UBICOMP2002), pp. 152-159, 2002.

[9] Stiefelhagen, R., Yang, J. and Waibel, A., Tracking Eyes and Monitoring Eye Gaze, In Proceedings of Workshop on Perceptive User Interfaces, pp. 98-100, 1997.

[10] IntelOpenCV, http://www.intel.com/research/mrl/research/opencv/