An Algorithm for Real-time Stereo Vision Implementation of Head Pose and Gaze Direction Measurement

Yoshio Matsumoto[†], **Alexander Zelinsky**[‡]
[†] Nara Institute of Science and Technology
8916-5 Takayamacho, Ikoma-city, Nara, Japan
yoshio@is.aist-nara.ac.jp
[‡] The Australian National University

Abstract

To build smart human interfaces, it is necessary for a system to know a user's intention and point of attention. Since the motion of a person's head pose and gaze direction are deeply related with his/her intention and attention, detection of such information can be utilized to build natural and intuitive interfaces.

In this paper, we describe our real-time stereo face tracking and gaze detection system to measure head pose and gaze direction simultaneously. The key aspect of our system is the use of real-time stereo vision together with a simple algorithm which is suitable for real-time processing. Since the 3D coordinates of the features on a face can be directly measured in our system, we can significantly simplify the algorithm for 3D model fitting to obtain the full 3D pose of the head compared with conventional systems that use monocular camera. Consequently we achieved a non-contact, passive, real-time, robust, accurate and compact measurement system for head pose and gaze direction.

1 Face and Gaze Detection for Visual Human Interfaces

Smart human interfaces need to know a user's intention and attention. For example, the direction of the gaze can be used for controlling the cursor on a monitor, and the motion of the head can be interpreted as a gesture such as "yes" or "no".

Several kinds of commercial products exist to detect a person's head position and orientation, such as magnetic sensors and link mechanisms. There are also several companies supporting products that perform eye gaze tracking. These products are generally highly accurate and reliable, however all requires either expensive hardware or artificial environments (cameras mounted on a helmet, infrared lighting, marking on the face etc). The discomfort and the restriction of the motion affects the person's behavior, which therefore makes it difficult to measure his/her natural behavior.

To solve this problem, many research results have been reported to visually detect the pose of a head [1, 2, 3, 4, 5, 6, 7]. Recent advances in hardware have allowed vision researchers to develop real-time face tracking systems. However most of these systems use a monocular vision. Recovering the 3D pose from a monocular image stream is known to be a difficult problem, and high accuracy as well as robustness are hard to be achieved. Therefore, some approaches can not compute the full 3D, 6DOF pose of the head, while other methods are not sufficiently accurate as a measurement system. Some researchers have also developed vision systems to passively detect gaze point [8, 9, 10, 11], however, none of which can measure the 3D vector of the gaze line.

In order to construct a system which observes a person without giving him/her any discomfort, it should satisfy the following requirements:

- non-contact
- passive
- real-time
- robust to occlusions, deformations and lighting fluctuations
- compact
- accurate
- able to detect head pose and a gaze direction simultaneously

Our system satisfies all these requirements by utilizing the following techniques:

- real-time stereo vision hardware using a field multiplexing device,
- image processing board with normalized correlation capability,
- 3D facial feature model and model fitting based on virtual springs,
- 3D eye model which assumes the eyeball to be a sphere.

The details of the hardware are described in Section 2, and the algorithm and implementation for face track-

[*] This research was done at The Australian National University.

ing and gaze detection are described in Section 3. Experimental results that show the real-time performance of the system are presented in Section 4. Finally the conclusions and a discussion of the future work are given in Section 5.

2 Hardware Configuration of Real-time Stereo Vision System

The hardware setup of our real-time stereo face tracking system is shown in **Figure 1** . We use a NTSC camera pair (SONY EVI-370DG × 2) to capture images of a person's face. The output video signals from the cameras are multiplexed into one video signal by the "field multiplexing technique"[12]. The multiplexed video stream is then fed into a vision processing board (Hitach IP5000), where the pose of the head and the direction of the gaze are calculated.

2.1 IP5000 Image Processing Board

The IP5000 is a PCI half-sized image processing board. It is connected to a NTSC camera source and a video output monitor. It is equipped with 40 frame memories of 512 × 512 pixels. The image processing LSI runs at 73.5[MHz] and provides a wide variety of fast image processing functions performed in hardware. These include binarization, convolution, filtering, labeling, histogram calculation, color extraction and normalized correlation. The main usage of this board in our system is the execution of normalized correlation for feature tracking and stereo matching.

2.2 Field Multiplexing Device

The field multiplexing is a technique used to generate a multiplexed video steam from two video streams in the analog phase. A diagram of the device is shown in **Figure 2** . The device takes two synchronized video steams as input into a video switching IC, and one of them is selected and output in every odd or even field.

Thus frequency of the switching is only 60[Hz], which makes the device easy and cheap to be implemented. A photo of the device is also shown in **Figure 2** . The size is less than 5[cm] square using only consumer electronic parts.

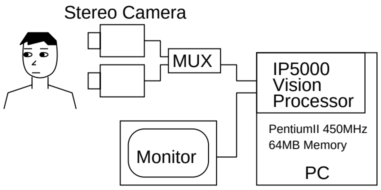

Fig. 1 : Hardware Configuration of Measurement System.

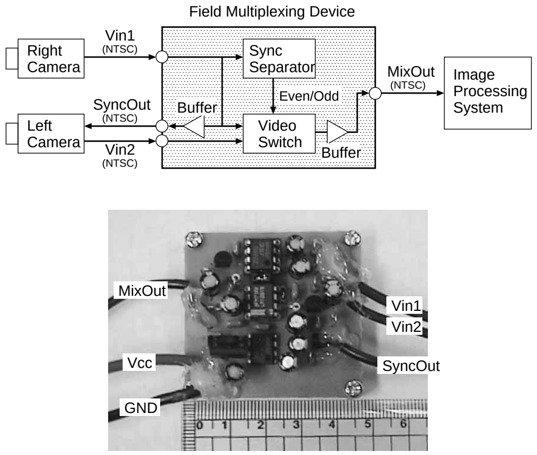

Fig. 2 : Block diagram and a photograph of the Field Multiplexing Device

The output video signal from the device contains a stereo image pair in the video frame with each image in half resolution in the vertical direction. The advantage of multiplexing video signals in the analog phase is that this technique can be applied to any vision system which takes a single video stream as input, and enables it to perform stereo vision processing. Since the multiplexed image is stored in a single video frame memory, stereo image processing can be performed within the memory. This means there is no overhead cost for image data transfer which is inevitable in stereo vision system with two image processing boards. Thus a system with a field multiplexing device can have a higher performance than a system with two boards.

3 Algorithm for Head Pose and Gaze Direction Measurement

The outline of the software configuration for face tracking and gaze detection is shown in **Figure 3** . It consists of three major parts, 1) Initialization, 2) Face Tracking and 3) Gaze Detection.

In the initialization stage, the system searches the face in the whole image using a 2D template of the whole face. After a face is found, the system starts face tracking where a 3D facial feature model is used to determine the 3D pose of the head. If the tracking of the face is not successful, the system regards the face to be lost and it jumps back to the initialization stage to find the face again. If the tracking of the face is successful, the system then calculates the direction of the gaze in the gaze detection stage. The 3D head pose and the 3D eye model are used to determine the 3D gaze vector. Finally, the system jumps back to the face tracking stage in the next frame.

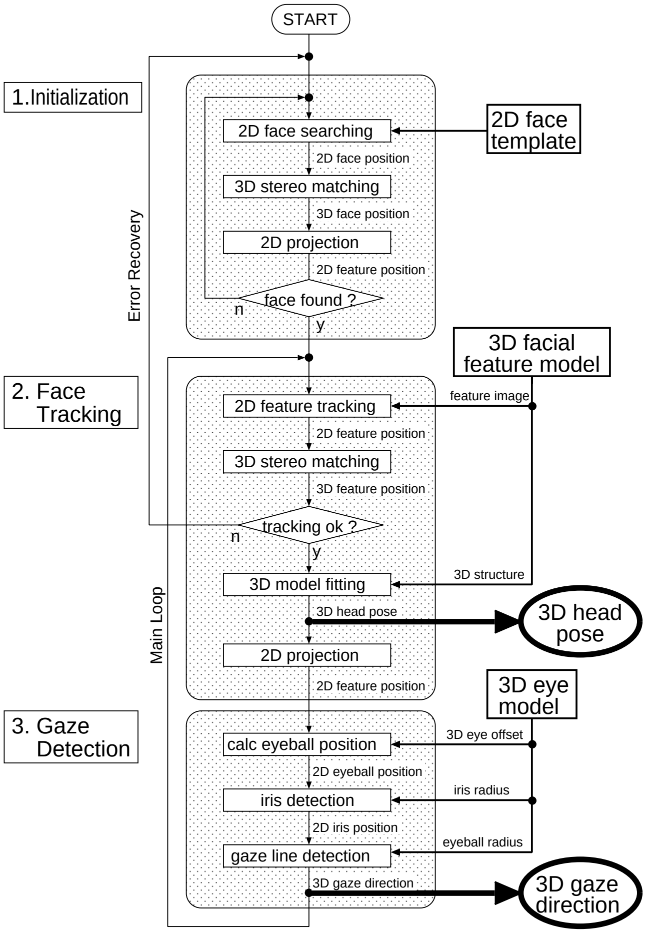

Fig. 3 : Software configuration.

3.1 Initialization

The feature tracking method described in this section uses only a small search area in the image. This enables real-time processing and continuously stable tracking. However, once the system fails to track the face, it is hard for the system to make a recovery by only using the local template matching scheme. Therefore a complementary method for finding the face in the image is necessary. This process is also used to initialize the position of the head at the beginning of the tracking. The whole face image shown in **Figure 5** is used in this process. In order to reduce the processing time, the template is stored in memory in low resolution. The live video streams are also reduced in resolution. The template is first searched in the right image, and then the matched image is searched in the left image. As a result, the rough 3D position of the face is determined and this is then used as the initial state of the face for the face tracking. This searching process takes about 100[ms].

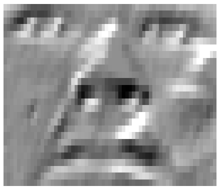

Fig. 4 : Whole face template.

3.2 Face Tracking

3.2.1 3D Facial Feature Model

The 3D facial feature model used in our stereo face tracking is composed of two components:

- images of the facial features,
- 3D coordinates of the facial features.

The facial features shown in **Figure 5** are defined as the corners of the eyes and the mouth. They can be regarded as patterns which are distinctive and suitable for tracking based on template matching. The facial feature model depends on each user, and can be built by simply selecting the feature position in an image with a mouse. The system then performs stereo matching to calculate the 3D coordinates for each feature.

3.2.2 3D Feature Tracking

In the 3D feature tracking stage, the 2D position of each feature in the previous frame is used to determine the search area in the current frame. The feature images stored in the 3D facial feature model are used as templates, and the stereo image pair are the search area. The 3D coordinates of each feature are acquired after stereo tracking. The processing time of the whole tracking process (i.e. feature tracking + stereo matching for six features) is approximately 10[ms] by IP5000.

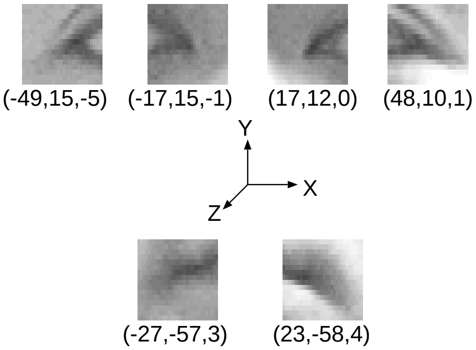

Fig. 5 : 3D facial feature model.

3.2.3 3D Model Fitting

Obtaining the best estimate of the head pose can be defined as a problem to determine the rotation matrix R and the translation vector t which minimize the squared fitting error E in the following equation:

$$E = \sum_{i=0}^{N-1} w_i(Rx_i + t - y_i)^T(Rx_i + t - y_i)$$

where N is the number of the features, x_i is the coordinate of a feature in the 3D feature model and y_i is the 3D measurement of a feature acquired in the 3D feature tracking and w_i is the weighting factor for each measurement. The correlation values obtained at the feature tracking and stereo matching are between 0 and 1, and they are multiplied and regarded as the weighting factor.

This problem can be solved using least squares. However, we adopted simpler gradient method using virtual springs, since we can safely assume that only a small displacement can occur between frames. The diagrams in **Figure 6** describe the model fitting method. In the real implementation all six features are used for fitting, however only three points are illustrated in the diagrams for simplicity. The basic idea of the model fitting is to move the model closer to the measurement iteratively while considering the reliability of the result of the 3D feature tracking. As stated above, we assume there can be only small displacements in terms of the position and the orientation, which are described as (Δx,Δy,Δz,$\Delta\phi$, $\Delta\theta$, $\Delta\varphi$) in **Figure 6** (1).

The position and the orientation acquired in the previous frame (at time t) are used to rotate and translate the measurement sets to move closer to the model as shown in **Figure 6** (2). After the rotation and translation, the measurements still have a small disparity to the model due to the motion which occurred during the interval Δt. Then the fine model fitting is performed. The product of the two correlation values are regarded as weighting factors which are between 0 and 1. This value is used as the stiffness of the springs between each feature in the model and the corresponding measurement as shown in **Figure 6** (3). The model is then rotated and translated gradually and iteratively to reduce the elastic energy of the springs, and R and t are updated to the best estimate. The processing time of this gradient model fitting takes less than 2[ms] on a PentiumII 450MHz computer.

Finally the 3D coordinates of each feature are adjusted to keep the consistency of the rigid body of the facial feature model, and they are projected back onto 2D image plane in order to update the search area for feature tracking in the next frame.

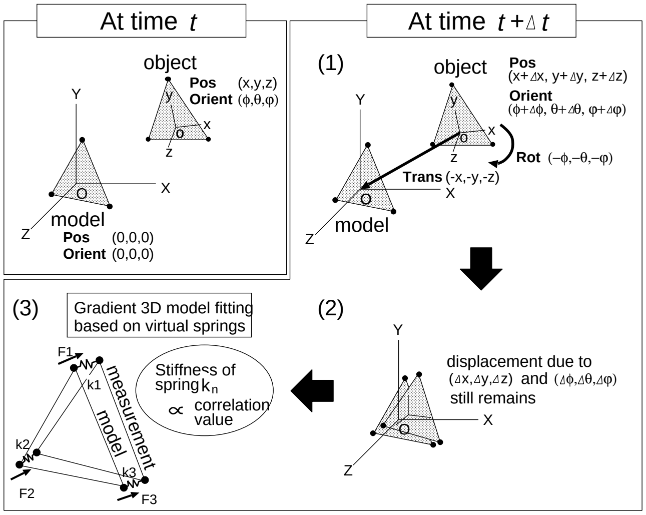

Fig. 6 : 3D model fitting algorithm.

3.3 Gaze Detection

In the modeling of the gaze line, the eyeballs are regarded as spheres. Gaze direction is determined based on both the pose of the head and the position of the irises of the eyes. The 3D eye model consists of following parameters:

- the relative position of the center of the eyeball respect to the head pose,
- radius of the eyeball,
- radius of the iris.

The relative position of the center of the eyeball is defined as a 3D vector from the mid-point of the corners of an eye to the center of the eyeball, and termed as an “offset vector.” The radius of the eyeball is a value around 13[mm], and the radius of the iris is a value around 7[mm]. These parameters are currently determined by the manual adjustment through a training sequence where the gaze point of a person is known.

Figure 7 illustrates the process to determine the 3D gaze direction. As shown in **Figure 7** (1), the 3D position of the eyeball can be determined from the pose of the head using the offset vector, although the eyeball center cannot be seen. By projecting the eyeball with known position and size back onto the image plane, the 2D appearance of the eyeball can be determined (**Figure 7** (2)). Next, the center of the iris is detected by using the circular Hough Transform as shown in **Figure 7** (3). Since the corner of the eyes are already known, the iris detection is executed on a small region between them, which typically takes about 10[ms].

The relationship between the iris center and eyeball center in the image plane defines the orientation of the gaze vector. **Figure 7** (4) indicates how to compute the horizontal angle of the gaze vector θ, and the vertical angle can be computed in the same manner. The modeling of the gaze direction in our system

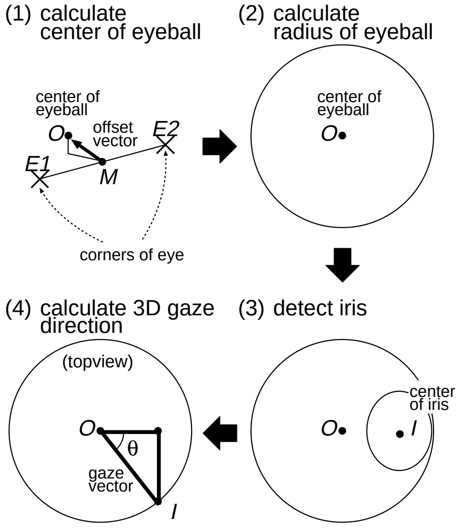

Fig. 7 : Modeling of gaze direction.

is quite straightforward, however, since it requires accurate head pose to determine the eyeball center, no other research has successfully adopted such simple modeling so far.

There are four eyes in total in stereo image pair, therefore four gaze direction are detected independently. However each measurement is not sufficiently accurate, mainly due to the resolution of the image. The field of view of the camera is set to capture the whole face in the image, then the width of an eye is only about 30[pixel] and the radius of the iris is only about 5[pixel] in a typical situation. Therefore it is hard to determine the "gaze point" in a 3D scene by calculating the intersection of the detected gaze lines. Therefore the those four vectors are currently averaged to generate a single gaze vector in order to reduce the effect of noises.

4 Experimental Results

4.1 Face Tracking

Some snapshots obtained during tracking experiments using our system are shown **Figure 8** . Images (1) and (2) in **Figure 8** show results when the face has rotations, while Image (3) shows the result when the face moves closer to the camera. The whole tracking process takes approximately 30[ms] which is well within the NTSC video frame rate. The accuracy of the tracking is approximately ±1[mm] in translation and ±1[deg] in rotation. In **Figure 8** Images (4),(5) and (6) show the results of tracking when there is some deformation of the facial features and partial occlusions of the face by a hand. The results indicate our tracking system works quite robustly in such situations due to the model fitting method. By utilizing the normalized correlation function on the IP5000, the tracking system is also tolerant to significant fluctuations in lighting.

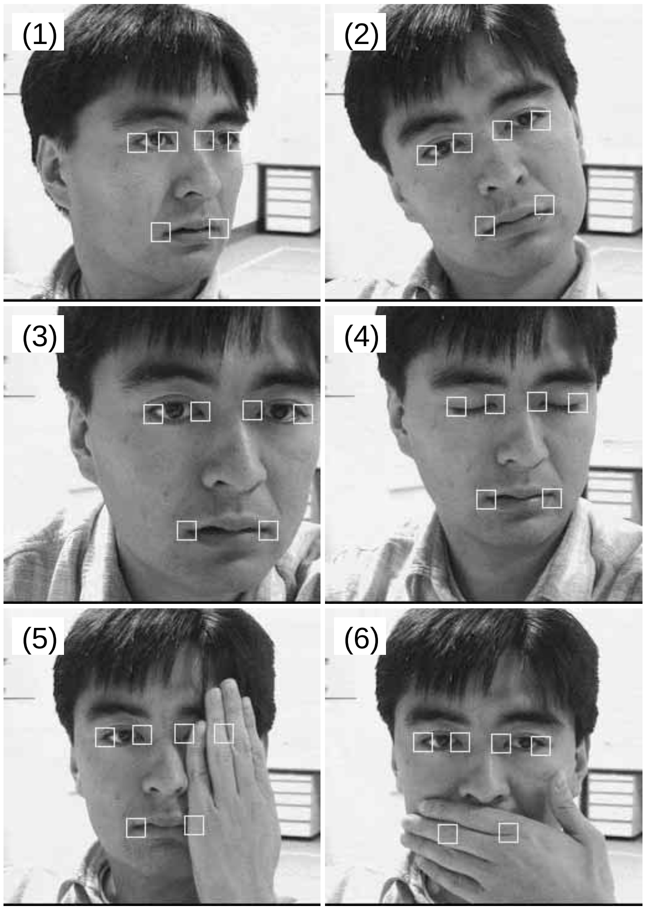

Fig. 8 : Result of face tracking at various situations.

4.2 Gaze Detection

Figure 9 shows some snapshots obtained in a real-time gaze detection experiment. The 3D gaze vectors are superimposed on the tracking result. The whole process including face tracking and gaze detection takes about 45[ms], thus the 3D gaze vector can be determined at 15[Hz].

The accuracy of the gaze direction are evaluated through a experiment using a board with markers, which is shown in **Figure 10** . The person sits 0.8[m] away from the camera pair. The distance between the markers which is 10[cm], which corresponds to 5.7[deg] in terms of the gaze direction. The results shown in **Figure 10** indicates the accuracy of the gaze vector is about 3[deg] in the worst case.

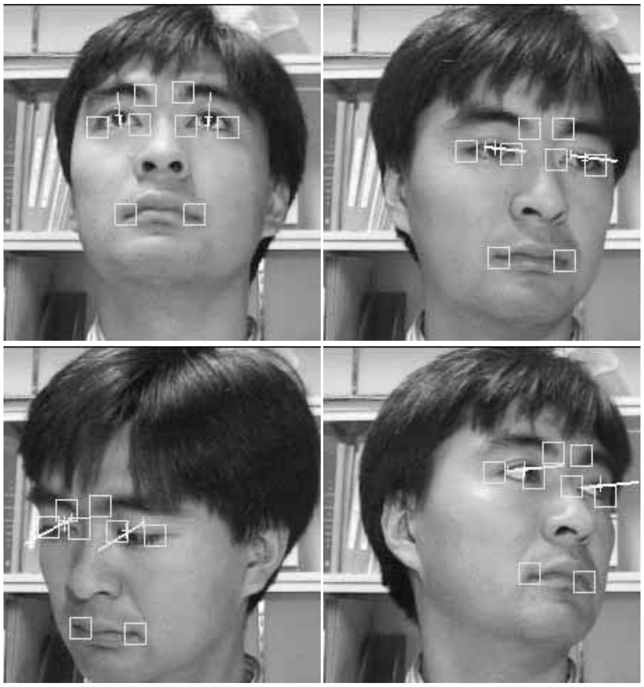

Fig. 9 : Result of detection of gaze direction.

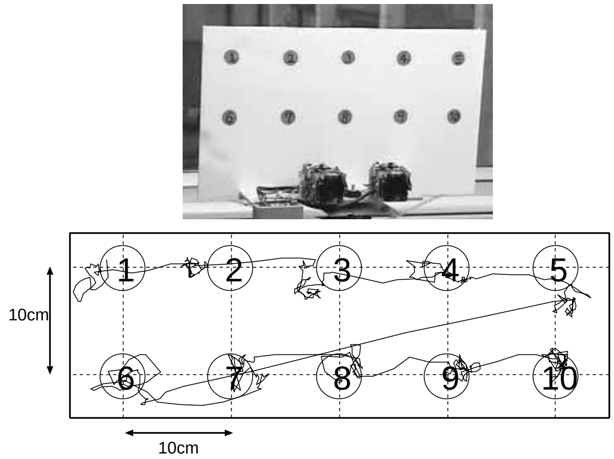

Fig. 10 : Result of accuracy assessment of gaze direction.

5 Conclusion

In this paper, a real-time implementation of our measurement system for head pose and gaze direction using stereo vision was presented. The system consists of a stereo camera pair and a standard PC equipped with an image processing board. The measurement system is (1) non-contact, (2) passive, (3) real-time and (4) accurate, all of which have not been able to be achieved by previous research results. The qualitative accuracy and robustness of the tracking is yet to be evaluated, however we believe that the performance of the system is quite high compared with existing systems.

This system can be applied to various targets, such as psychological experiments, ergonomic designing, products for the disabled and the amusement industry. In our future work, we will evaluate the accuracy of the head pose and the gaze direction. We also aim to improve the accuracy and processing speed of the gaze detection.

References

[1] A.Azarbayejani, T.Starner, B.Horowitz, and A.Pentland. Visually controlled graphics. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(6):602–605, 1993.

[2] A.Zelinsky and J.Heinzmann. Real-time Visual Recognition of Facial Gestures for Human Computer Interaction. In *Proc. of the Int. Conf. on Automatic Face and Gesture Recognition*, pages 351–356, 1996.

[3] P.Ballard and G.C.Stockman. Controlling a Computer via Facial Aspect. *IEEE Trans. Sys. Man and Cybernetics*, 25(4):669–677, 1995.

[4] Black and Yaccob. Tracking and Recognizing Rigid and Non-rigid Facial Motions Using Parametric Models of Image Motion. In *Proc. of Int. Conf. on Computer Vision (ICCV'95)*, pages 374–381, 1995.

[5] S.Birchfield and C.Tomasi. Elliptical Head Tracking Using Intensity Gradients and Color Histograms". In *Proc. of Computer Vision and Pattern Recognition (CVPR'98)*, 1998.

[6] A.Gee and R.Cipolla. Fast Visual Tracking by Temporal Consensus. *Image and Vision Computing*, 14(2):105–114, 1996.

[7] Kentaro Toyama. Look, Ma – No Hands! Hands-Free Corsor Control with Real-time 3D Face Tracking. In *Proc. of Workshop on Perceptual User Interface (PUI'98)*, 1998.

[8] Shumeet Baluja and Dean Pomerleau. Non-Intrusive Gaze Tracking Using Artificial Neural Networks. Technical Report CMU-CS-94-102, CMU, 1994.

[9] C.Colombo, S.Andronico, and P.Dario. Prototype of vision-based gaze-driven man-machine interface. In *Proc. of IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, pages 188–192, 1995.

[10] J.Heinzmann and A.Zelinsky. 3-D Facial Pose and Gaze Point Estimation using a Robust Real-Time Tracking Paradigm. In *Proc. of the Int. Conf. on Automatic Face and Gesture Recognition*, 1998.

[11] R.Stiefelhagan, J.Yang, and A.Waibel. Tracking Eyes and Monitoring Eye Gaze. In *Proc. of Workshop on Perceptual User Interface (PUI'97)*, 1997.

[12] Y. Matsutmoto, T. Shibata, K. Sakai, M. Inaba, and H. Inoue. Real-time Color Stereo Vision System for a Mobile Robot based on Field Multiplexing. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 1934–1939, 1997.