

Gaze Detection by Wide and Narrow View Stereo Camera

Kang Ryoung Park

Division of Media Technology, SangMyung University, 7 Hongji-Dong, JongRo-Gu, Seoul, Republic of Korea

Abstract. Human gaze is very important information for the interaction with the computer. In this paper, we propose the new gaze detection system with a wide and a narrow view stereo camera. In order to locate the user's eye position accurately, the narrow-view camera has the functionalities of auto P/T/Z/F based on the detected 3D eye positions from the wide view camera. In addition, we use the IR-LED illuminators for wide and narrow view camera, which can ease the detecting of facial features, pupil and iris position. The eye gaze position on a monitor is computed by a multi-layered perceptron with a limited logarithm function. Experimental results show that the gaze detection error between the computed positions and the real ones is about 2.89 cm of RMS error.

Keywords: Gaze Detection, Dual Cameras, Dual IR-LED Illuminators

1 Introduction

Human gaze is very important information for the interaction with the computer including many applications such as the view control in 3D simulation programs, virtual reality and video conferencing. In addition, they can help the handicapped to use computers and are also useful for those whose hands are busy controlling other menus on the monitor[19]. Most Previous studies were focused on 2D/3D head rotation/translation estimation[2][15], the facial gaze detection[3-9][16][17][19][23] and the eye gaze detection[10-14][18][24-29]. Recently, the gaze detection considering both head and eye movement has been researched. Rikert et al.[9]'s method has the constraints that the Z distance between a face and the monitor must be maintained unchanged during training and testing procedures, which can give much inconvenience to user. In the methods of [11][13][14][16][17], a pair of glasses having marking points is required to detect facial features, which can be also inconvenient to a user. The researches of [3][4][20] show the gaze detection methods only considering head movements and have the limits that the gaze errors are increased in case that the eye movements happen. To overcome such problems, the research of [21] shows the gaze detection considering both head and eye movements, but uses only one wide view camera, which can capture the whole face of user. In such case, the eye image resolution is too low and the fine movements of user's eye cannot be exactly detected. To overcome above problems, we propose the new method of computing

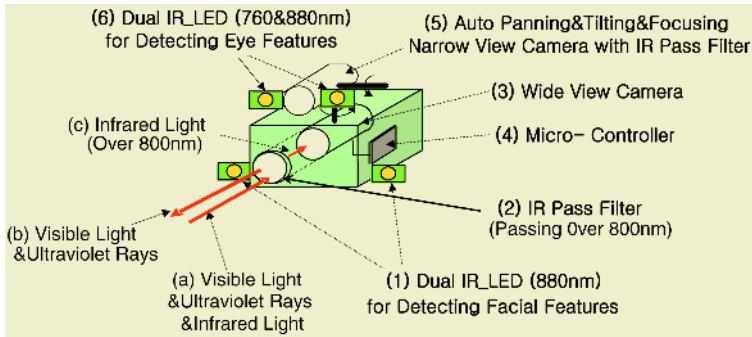


Fig. 1. The gaze detecting system

gaze position. In order to implement practical gaze system based on our method, we use dual cameras (a wide view and a narrow view camera). In order to track the positions of user's eye changed by head movements, the narrow view camera has the functionalities of auto focusing/panning/tilting based on the detected 3D eye positions from the wide view camera. In addition, we use IR-LED illuminators for wide and narrow view camera, which can ease the detecting of facial features, pupil and iris position. To overcome the problem of specular reflection on glasses by illuminator, we use dual IR-LED illuminators for wide and narrow view camera.

2 Localization of Facial Features in Wide View Image

In order to detect gaze position on a monitor, we first locate facial features (both eye centers, eye corners, nostrils) in wide view images. There have been many researches for detecting facial features. One of them is to use facial skin color[22], but their performance can be affected by the environmental light or race, etc. To overcome such problems and detect the facial features robustly in any environment, we use the method of detecting specular reflection on the eyes. For that, we implement the gaze detection system as shown in Fig. 1.

As shown in Fig. 1, the IR-LED(1) is used to make the specular reflections on eyes. The IR pass filter(2) in front of camera lens can only pass the infrared light (over 800 nm) and the brightness of input image is only affected by the IR-LED(1) excluding external illumination. We use a wide view(3) and a narrow view(5) CCD camera (interlaced scan mode). The reason of using IR-LED(1) of 880nm is that human eye can only perceive the visible and the near infrared light (below about 880nm) and our illuminators do not make dazzling to user's eye, consequently. When a user starts our gaze detection S/W in PC, the starting signal is transmitted to the micro-controller(4) in camera via the RS-232C. Then, the micro-controller turns on the illuminator(1) synchronized with the even field of CCD signal and turns off it synchronized with the next odd field of CCD signal, successively[21]. From that, we can get a difference image between the even and the odd image and the specular reflection points on both

eyes can be easily detected because their image gray level are higher than other regions[21]. Around the detected corneal specular reflection points, we determine the eye candidate region of 30×30 pixels and locate the accurate eye (iris) center by the circular edge detection method[30]. Because the eye localization is performed in the restricted region, it can be done in real-time (below 3 ms in Pentium-III 866MHz). After locating the eye center, we detect the eye corner by using eye corner shape template and SVM (Support Vector Machine)[21]. We get 2000 successive image frames (100 frames \times 20 persons in various sitting positions) for SVM training. From that, 8000 eye corner data (4 eye corners \times 2000 images) are obtained and additional 1000 (50 frames \times 20 persons in various sitting positions) images are used for testing. Experimental results show the classification error for training data is 0.11% (9/8000) and that for testing data is 0.2% (8/4000) and our algorithm is also valid for the users with glasses or contact lens. In comparing experiments, MLP (Multi-Layered Perceptron) shows the worse performance (the error of 1.58% for training data and 3.1% for testing data). The classification time of SVM is so small as like 8 ms in Pentium-III 866MHz. After locating eye centers and eye corners, the positions of nostrils can be detected by anthropometric constraints in a face and SVM. In order to reduce the effect by the facial expression change, we do not use the lip corners for gaze detection. Experimental results show that RMS error between the detected feature positions and the actual positions (manually detected positions) are 1 pixel (of both eye centers), 2 pixels (of both eye corners) and 4 pixels (of both nostrils) in 640×480 pixels image. From them, we use 5 feature points (left/right eye corners of left eye, left/right eye corners of right eye, nostril center) in order to detect facial gaze position.

3 4 Steps for Computing Facial Gaze Position

After feature detection, we take 4 steps in order to compute a gaze position on a monitor[3][4][21]. At the 1st step, when a user gazes at 5 known positions on a monitor, the 3D positions (X, Y, Z) of initial 5 feature points (detected in the section 2) are computed automatically[3][4]. At the 2nd step and 3rd step, when the user rotates/translates his head in order to gaze at one position on a monitor, the new (changed) 3D positions of those 5 features can be computed from 3D motion estimation. Considering many limitations of previous motion estimation researches, we use the EKF (Extended Kalman Filtering)[2] for 3D motion estimation and the new 3D positions of those features can be computed by the EKF and affine transform[3][21]. The estimation accuracy of the EKF is compared with 3D position tracking sensor[32]. Experimental results show the RMS errors are about 1.4 cm and 2.98° in translation and rotation. In addition, the experimental results show that the RMS error of between the changed(estimated) 3D positions of 5 features and the actual ones (measured by 3D position tracking sensor) is 1.15 cm (0.64cm in X axis, 0.5cm in Y axis, 0.81cm in Z axis) for 20 person data which were used for testing the feature detection performance. At the 4th step, one facial plane is determined from the new (changed) 3D posi-

tions of the 5 features and the normal vector (whose origin exists in the middle of the forehead) of the plane shows a gaze vector by head (facial) movements. The gaze position on a monitor is the intersection position between a monitor and the gaze vector[3][4] as shown in Fig. 2(b).

4 Auto Panning/Tilting/Focusing of Narrow View Camera

Based on the new (changed) 3D positions of the 5 feature points (which are computed at the 2nd and 3rd step as mentioned in section 3), we can pan and tilt the narrow view camera in order to capture the eye image. For that, we also perform the coordinate conversion between monitor and narrow view camera using the internal/external camera parameters, which are obtained at initial calibration stage. Such calibration method is same to that between the wide view camera and the monitor. Detail accounts can be referred in [3]. When the user rotates his head severely, one of his eyes may disappear in camera view. So, we track only one visible eye with auto panning/tilting narrow view camera. Conventional narrow view camera has small DOF (Depth of Field) and there is the limitation of increasing the DOF with the fixed focal camera. So, we use the auto focusing narrow view camera in order to capture clear eye image. For auto focusing, the Z distance between the eye and the camera is required and we can obtain the Z distance at the 2nd and 3rd step (as mentioned in section 3). In order to compensate the focusing error due to the inaccurate Z distance measure, we use an additional focus quality checking algorithm. That is, the auto focusing for eye image is accomplished based on the computed Z distance and the captured eye image is transmitted to PC. With that, the focus quality checking algorithm computes the focus quality. If the quality does not meet our threshold (70 of the range (0 ~ 100)), then we perform additional focusing process by sending the moving command of focus lens to camera micro-controller. In this stage, we should consider the specular reflection on glasses. The surface of glasses can make the specular reflection, which can hide the whole eye image. In such case, the eye region is not detected and we cannot compute the eye gaze position. So, we use dual IR-LED illuminators like Fig. 1(6). When the large specular reflection happens from one illuminator (right or left illuminator), then it can be detected from image. As mentioned in section 2, the NTSC analog level of specular reflection region is higher than any other region and they can be detected by changing decoder brightness setting. When the large specular region proves to exist with the changed decoder brightness value, then our gaze detection system change the illuminator (from left to right or right to left) and the specular reflection on glasses does not happen, consequently.

5 Localization of Eye Features in Narrow View Image

After we get the focused eye image, we perform the localization of eye features. We detect $P_1 \sim P'_4$ in right eye image and also detect $P_5 \sim P'_8$ in left eye image for

computing eye gaze detection. Here, the P_1 and P'_1 show the pupil center and the P_2 and P'_2 does the iris center. J. Wang et al.[1] uses the method that detects the iris outer boundary by vertical edge operator, morphological “open” operation and elliptical fitting. However, the upper and lower region of iris outer boundary tend to be covered by eyelid and inaccurate iris elliptical fitting happens due to the lack of iris boundary pixels. In addition, their method computes eye gaze position by checking the shape change of iris when a user gazes at monitor positions. However, our experimental results show that the shape change amount of iris is very small and it is difficult to detect the accurate eye gaze position only by that information. So, we use the positional information of both pupil and iris. Also, we use the information of shape change of pupil, which does not tend to be covered by eyelid. In general, the IR-LED of short wavelength (700nm \sim 800nm) makes the high contrast between iris and sclera. On the other hand, that of long wavelength (800nm \sim 900nm) makes the high contrast between pupil and iris. Based on that, we use the IR-LED illuminator of multi-wavelength (760nm and 880nm) as shown in Fig. 1(6). The shapes of iris and pupil are almost ellipse, when the user gazes at a side position of monitor. So, the method of circular edge detection[30] cannot be used. Instead, we use the canny edge operator to extract edge components and a 2D edge-based elliptical Hough transform[31]. From that, we can get the center positions and the major/minor axes of iris/pupil ellipses. In order to detect the eye corner position, we detect the eyelid. That is because the upper and lower eyelids meet on two eye corner positions. To extract the eyelid region, we use the region-based template deformation and masking method. In detail, we make the eyelid edge image with canny edge operator and apply the deformable template as the eyelid mask. Here, we use 2 deformable templates (parabolic shape) for upper and lower eyelid detection. From that, we can detect the accurate eye corners. Experimental results show that RMS errors between the detected eye feature positions and the actual ones (manually detected) are 2 pixels (of iris center), 1 pixel (of pupil center), 4 pixels (of left eye corner) and 4 pixels (of right eye corner). Based on the detected eye features, we select the 22 feature values ($f_1 \sim f_{11}$ are used in case that right eye image can be captured by narrow view camera and $f_{12} \sim f_{22}$ are used in case that left eye image can be captured). With those feature values, we can compute eye gaze position on a monitor as shown in Fig. 2(a). Detail accounts are shown in section 6.

6 Detecting the Gaze Position on a Monitor

In section 3, we explain the gaze detection method only considering head movement. As mentioned before, when a user gazes at a monitor position, both the head and eyes tend to be moved simultaneously. So, we compute the additional eye gaze position with the detected 22 feature values (as mentioned in section 5) and a neural network (multi-layered perceptron) as shown in Fig. 2(a). Here, the input values for neural network are normalized by the distance between the iris/pupil center and the eye corner, which are obtained in case of gazing at monitor center. That is because we do not use a zoom lens in our camera. That

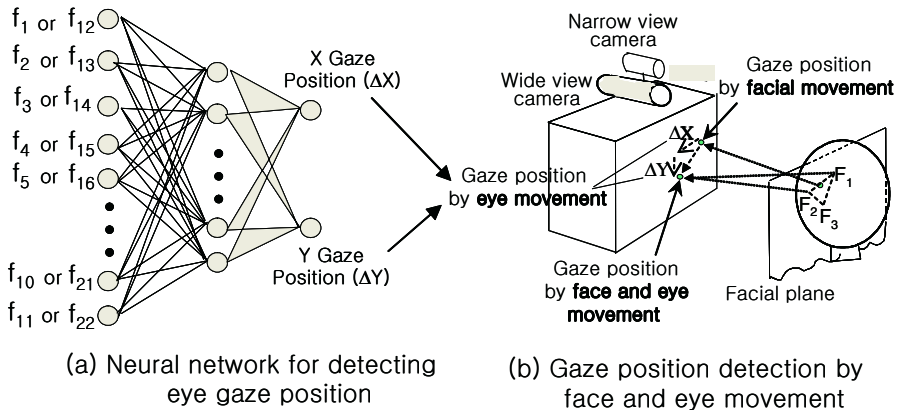


Fig. 2. The neural network for eye gaze detection and gaze position detection by face and eye movement

is, the more the user approaches the monitor (camera), the larger the eye size becomes and the farther the distance between the pupil/iris and the eye corner becomes, consequently. After detecting eye gaze position based on the neural network, we can determine a final gaze position on a monitor by head and eye movements based on the vector summation of each gaze position (face and eye gaze) as shown in Fig. 2(b).

7 Performance Evaluations

The gaze detection error of the proposed method is compared to that of our previous methods[3][4][19][21] as shown in Table 1. The researches[3][4] compute facial gaze position not considering the eye movements. The research[19] calculates the gaze position by mapping the 2D facial feature position into the monitor gaze position by linear interpolation or neural network without 3D computation and considering eye movements. The method[21] computes the gaze positions considering both head and eye movements, but uses only one wide view camera. The test data are acquired when 10 users gaze at 23 gaze positions on a 19" monitor. Here, the gaze error is the RMS error between the actual gaze positions and the computed ones. Shown in Table 1, the gaze errors are calculated in two cases. The case I shows that gaze error about test data including only head movements and the case II does that the gaze error including head and eye movements.

Shown in Table 1, the gaze error of the proposed method is the smallest in any case. At the 2nd experiment, the points of radius 5 pixels are spaced vertically and horizontally at 1.5" intervals on a 19" monitor with the screen resolution of 1280×1024 pixels as such Rikert's research[9]. The RMS error between the real and calculated gaze position is 2.85 cm and it is much superior to Rikert's method (almost 5.08 cm). Our gaze error is correspondent to the angular error of

Table 1. Gaze error about test data (cm)

Method	Linear interpol.[19]	Single neural net[19]	Combined neural nets[19]	[3] method	[4] method	[21] method	Proposed method
case I	5.1	4.23	4.48	5.35	5.21	3.40	2.24
case II	11.8	11.32	8.87	7.45	6.29	4.8	2.89

2.29 degrees on X axis and 2.31 degrees on Y axis. In addition, we tested the gaze errors according to the Z distance (55, 60, 65cm). The RMS errors are 2.81cm at 55cm, 2.85cm at 60cm, 2.92cm at 65cm. It shows that the performance of our method is not affected by the user's Z position. Last experiment for processing time shows that our gaze detection process takes about 500ms in Pentium-III 866MHz and it is much smaller than Rikert's method (1 minute in alphastation 333MHz). Our system only requires the user to gaze at 5 known monitor positions at the initial calibration stage (as shown in the section 3) and can track/compute the user's gaze position without any user's intervention at real-time speed. The research[1] shows the angular error of below 1 degree, but their method supposes that they know the 3D distance between two eyes and that between both lip corners and there is no individual variation for the 3D distances. In addition, they suppose that they know the 3D diameter of eye ball and there is no individual variation for that. However, our preliminary experiments show that there are much individual variations for the 3D distances/3D diameter and such cases can increase much gaze errors (the angular error of more than 4 degree).

8 Conclusions

This paper describes a new gaze detecting method. In future works, we have plans to research the method of capturing higher resolution eye image with zoom lens and it will increase the accuracy of final gaze detection. In addition, the method to increase the auto panning/tilting/focusing speed of narrow view camera should be researched to decrease total processing time.

References

1. J. Wang and E. Sung, 2002. Study on Eye Gaze Estimation, IEEE Trans. on SMC, Vol. 32, No. 3, pp.332-350
2. A. Azarbayejani., 1993, Visually Controlled Graphics. IEEE Trans. PAMI, Vol. 15, No. 6, pp. 602-605
3. K. R. Park et al., Apr 2000, Gaze Point Detection by Computing the 3D Positions and 3D Motions of Face, IEICE Trans. Inf.&Syst.,Vol. E.83-D, No.4, pp.884-894
4. K. R. Park et al., Oct 1999, Gaze Detection by Estimating the Depth and 3D Motions of Facial Features in Monocular Images, IEICE Trans. Fundamentals, Vol. E.82-A, No. 10, pp. 2274-2284
5. K. OHMURA et al., 1989. Pointing Operation Using Detection of Face Direction from a Single View. IEICE Trans. Inf.&Syst., Vol. J72-D-II, No.9, pp. 1441-1447

6. P. Ballard et al., 1995. Controlling a Computer via Facial Aspect. *IEEE Trans. on SMC*, Vol. 25, No. 4, pp. 669-677
7. A. Gee et al., 1996. Fast visual tracking by temporal consensus, *Image and Vision Computing*. Vol. 14, pp. 105-114
8. J. Heinzmann et al., 1998. 3D Facial Pose and Gaze Point Estimation using a Robust Real-Time Tracking Paradigm. *Proceedings of ICAFGR*, pp. 142-147
9. T. Rikert, 1998. Gaze Estimation using Morphable Models. *ICAFGR*, pp.436-441
10. A.Ali-A-L et al., 1997, Man-machine Interface through Eyeball Direction of Gaze. *Proc. of the Southeastern Symposium on System Theory*, pp. 478-82
11. A. Tomono et al., 1994. Eye Tracking Method Using an Image Pickup Apparatus. *European Patent Specification-94101635*
12. Eyemark Recorder Model EMR-NC, NAC Image Technology Cooperation
13. Porrill-J et al., Jan 1999, Robust and Optimal Use of Information in Stereo Vision. *Nature*. vol.397, no.6714, pp.63-6
14. Varchmin-AC et al., 1998, Image based Recognition of Gaze Direction Using Adaptive Methods. *Gesture and Sign Language in Human-Computer Interaction*. *Int. Gesture Workshop Proc. Berlin, Germany*, pp. 245-57.
15. J. Heinzmann et al., 1997. Robust Real-time Face Tracking and Gesture Recognition. *Proc. of the IJCAI*, Vol. 2, pp. 1525-1530
16. Matsumoto-Y, et al., 2000, An Algorithm for Real-time Stereo Vision Implementation of Head Pose and Gaze Direction Measurement. *Proc. the ICAFGR*. pp. 499-504
17. Newman-R et al., 2000, Real-time Stereo Tracking for Head Pose and Gaze Estimation. *Proceedings the 4th ICAFGR 2000*. pp. 122-8
18. Betke-M et al., 1999, Gaze Detection via Self-organizing Gray-scale Units. *Proc. Int. Workshop on Recog., Analy., and Tracking of Faces and Gestures in Real-Time System*. pp. 70-6
19. K. R. Park et al., 2000. Intelligent Process Control via Gaze Detection Technology. *EAAI*, Vol. 13, No. 5, pp. 577-587
20. K. R. Park et al., 2002. Gaze Position Detection by Computing the 3 Dimensional Facial Positions and Motions. *Pattern Recognition*, Vol. 35, No.11, pp. 2559-2569
21. K. R. Park et al., 2002, Facial and Eye Gaze detection. *LNCS*, Vol.2525, pp. 368-376
22. Jie Yang and Alex Waibel, A Real-time Face Tracker, *Proceedings of WACV'96*, pp. 142-147
23. Y. Matsumoto, 2000. An Algorithm for Real-time Stereo Vision Implementation of Head Pose and Gaze Direction Measurement, *ICFGR*, pp.499-505
24. <http://www.iscaninc.com>
25. <http://www.seeingmachines.com>
26. B Wolfe, D. Eichmann, 1997. A Neural Network Approach to Tracking Eye Position, *International Journal Human Computer Interaction*, Vol. 9, No.1, pp. 59-79
27. David Beymer and Myron Flickner, 2003. Eye Gaze Tracking Using an Active Stereo Head, *IEEE Computer Vision and Pattern Recognition*
28. J. Zhu et al., 2002. Subpixel Eye Gaze Tracking, *International Conference on Face and Gesture Recognition*
29. R. Stiefelhagen, J. Yang, and A. Waibel, 1997. Tracking Eyes and Monitoring Eye Gaze, *Proceedings of Workshop on Perceptual User Interfaces*, pp. 98-100
30. J. Daugman, 2003. The Importance of Being Random: Statistical Principles of Iris Recognition, *Pattern Recognition*, vol. 36, no. 2, pp. 279-291
31. Ramesh Jain, 1995, *Machine Vision*, McGraw-Hill International Edition
32. <http://www.polhemus.com>