

Non-intrusive Eye and Gaze Tracking for Natural Human Computer Interaction

QIANG JI & ZHIWEI ZHU

Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, JEC 6604, Troy, New York, USA, 12180-3590

Keywords: Eye Tracking, Gaze Estimation, Human Computer Interaction

1. Introduction

Gaze determines the user's current line of sight or point of fixation. The fixation point is defined as the intersection of the line of sight with the surface of the object being viewed (such as the screen). Gaze may be used to interpret the user's intention for non-command interactions and to enable (fixation dependent) accommodation and dynamic depth of focus. The potential benefits for incorporating eye movements into the interaction between humans and computers are numerous. For example, knowing the location of a user's gaze may help a computer to interpret a user's request and possibly enable a computer to ascertain some cognitive states of the user, such as confusion or fatigue.

The direction of the eye gaze can express the interests of the user; it is a potential porthole into the current cognitive processes. Communication through the direction of the eyes is faster than any other mode of human communication. In addition, real time monitoring of gaze position permits the introduction of display changes that are contingent on the spatial or temporal characteristics of eye movements. Such methodology is referred to as the gaze contingent display paradigm. For example, gaze may be used to determine one's fixation on the screen, which can then be used to infer the information of interest to the user. Appropriate actions can then be taken such as increasing the resolution or increasing the size of the region where the user fixates. Another example is to economize on bandwidth by putting high-resolution information only where the user is currently looking.

Gaze tracking is therefore important for Human Computer Interaction (HCI). Numerous techniques have been developed including some commercial eye gaze trackers. Basically, they can be divided into video-based techniques and non-video-

based techniques. Usually, non-video-based methods (Bour, 1997) use some special contacting devices attached to the skin or eye to obtain the user's gaze. So, they are intrusive and interfere with the user. Recently, using a non-contacting video camera together with a set of techniques, numerous video-based methods have been presented. Compared with non-video-based gaze tracking methods, video-based gaze tracking methods have the advantage of being unobtrusiveness and comfortable during the process of gaze estimation. We will concentrate on the video-based approaches in this paper.

The direction of a person's gaze is determined by two factors: the orientation of the face (face pose), and the orientation of eye (eye gaze). Face pose determines the global direction of the gaze, while eye gaze determines the local direction of the gaze. Global gaze and local gaze together determine the final gaze of the person. According to these two aspects of gaze information, video-based gaze estimation approaches can be partitioned into a head-based approach, an ocular-based approach and a combined head and eye-based approach.

The head-based approach determines user's gaze based on the head orientation. In Rae & Ritter (1998), a set of Gabor filters is applied locally to the image region that includes the face. This results in a feature vector to train a neural network to predict the two neck angles, pan and tilt, providing the desired information about head orientation. Mukesh & Ji (2001) introduced a robust method for discriminating 3D face pose (face orientation) from a video sequence featuring views of a human head under variable lighting and facial expression conditions. Gee & Cipoll (1994) estimated the user's head orientation as the user's gaze direction from a single, monocular view of a face by ignoring the eyeball's rotation. Our recent efforts (Ji, 2002; Ji & Yang, 2001) in this area led to the development of several techniques for face pose estimation. Gaze estimation by head orientation, however, only provides a global gaze since one's gaze can still vary considerably given the head orientation. By looking solely at the head movements, the accuracy of the user's gaze is traded for flexibility.

The ocular-based approach estimates gaze by establishing the relationship between gaze and the geometric properties of the iris or pupil. The special character of the iris structure, namely the transition from white-to-dark then dark-to-white, makes it possible to segment the iris from the eye region reliably. The special bright pupil effect under IR illumination makes pupil segmentation very robust and effective. Specifically, the iris-based gaze estimation approach computes gaze by determining the iris location from the iris' shape distortions, while the pupil-based approach determines gaze based on the relative spatial positions between pupil and the glint (cornea reflection). So far, the most common approach for ocular-based gaze estimation is based on the relative position between pupil and the glint (cornea reflection) on the cornea of the eye (Ebisawa, 1995; Ebisawa, 1998; Hutchinson, 1988; Hutchinson et al., 1989; Ohno et al., 2002; Koons & Flickner; Morimoto et al., 1999). Assuming a static head, methods based on this idea use the glint as a reference point, thus the vector from the glint to the center of the pupil is used to infer the gaze direction, assuming the existence of a simple analytical function that maps glint vector to gaze. While contact-free and non-intrusive, these methods work well only for a static head, which is a rather restrictive constraint on the part of the user. Even a chin rest is used to maintain the head still because minor head movement can cause these techniques to fail. This poses a significant hurdle to natural human computer

interaction. Another serious problem with the existing eye and gaze tracking systems is the need to perform a rather cumbersome calibration process for each individual. For example, in Morimoto et al. (1999), nine points are arranged in a 3×3 grid on the screen, and the user is asked to fixate his/her gaze on a certain target point one by one. For each fixation, the pupil-glint vector and the corresponding screen coordinate are obtained, and a simple second order polynomial transformation is used to obtain the mapping relationship between the pupil-glint vector and the screen coordinates. Such a simple calibration, however, is not acceptable for human computer interaction in a natural environment.

In view of these limitations, in this paper, we present a gaze estimation approach that accounts for both the local gaze computed from the ocular parameters and the global gaze computed from the head pose. The global gaze (face pose) and local gaze (eye gaze) are combined together to obtain the precise gaze information of the user. A general approach that combines head pose information with eye gaze information to perform gaze estimation is proposed. Our approach allows natural head movement while still estimating gaze accurately. Another effort is to make the gaze estimation calibration free. New users or the existing users who have moved, do not need to undergo a personal gaze calibration before using the gaze tracker. Therefore, the gaze tracker can perform robustly and accurately without calibration and under natural head movements. An overview of the major components of our gaze tracking algorithm is shown in Figure 1.

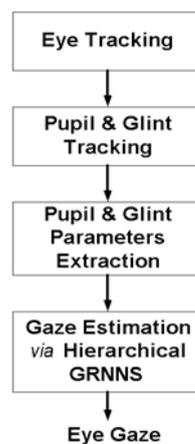


Figure 1: Major components of the proposed system

2. Eye Tracking

Gaze tracking starts with eye tracking. For eye tracking, we track pupils instead. We use infrared LEDs that operate at a power of 32mW in a wavelength band 40nm wide at a nominal wavelength of 880nm. As in Ji & Yang (2001), we obtain a dark and a bright pupil image by illuminating the eyes with IR LEDs located off (the outer IR ring) and on the optical axis (the inner IR ring), respectively. To further improve the quality of the image and to minimize interference from light sources other than the IR illuminator, we use an optical band-pass filter which has a wavelength pass band only 10nm wide. The filter has increased the signal-to-noise ratio significantly, compared with the case without using the filter. Figure 2 illustrates the IR illuminator consisting of two concentric IR rings and the band-pass filter.

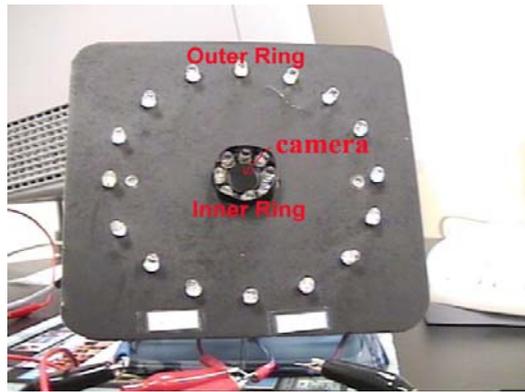


Figure 2: Hardware setup: the camera with an active IR illuminator.

Pupils detection and tracking start with pupils detection in the initial frames, followed by tracking. Figure 3 summarizes our pupil detection and tracking algorithm. Pupil detection is accomplished based on both the intensity of the pupils (the bright and dark pupils as shown in Figure 5) and on the appearance of the eyes using the support vector machine. The use of support vector machine (SVM) avoids falsely identifying a bright region as a pupil. Specifically, candidates of pupils are first detected from the difference image which results from subtracting the dark pupil image from the bright pupil image. The pupil candidates are then validated using SVM to remove spurious pupil candidates. Given the detected pupils, pupils in the subsequent frames can be detected efficiently via tracking. Kalman filtering is used since it allows pupils' positions in the previous frame to predict pupils' positions in the current frame, therefore greatly limiting the search space. Kalman filtering tracking based on pupil intensity is therefore implemented. To avoid Kalman filtering going awry due to the use of only intensity, Kalman filtering is augmented by mean-shift tracking, which tracks an object based on its intensity distribution. Details on our eye detection and tracking may be found in Zhu et al. (2002).

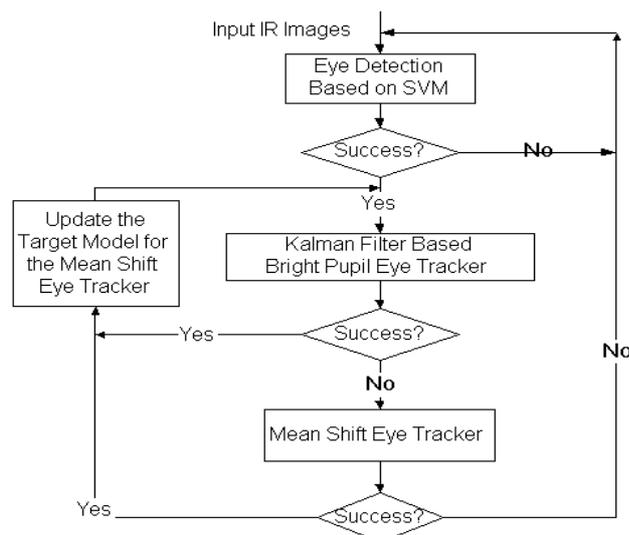


Figure 3: Flowchart of our pupil detection and tracking algorithm

3. Gaze Determination and Tracking

Our gaze estimation algorithm consists of three parts: pupil-glint detection and tracking, gaze calibration, and gaze mapping. For this research, one's gaze is quantized into 8 regions on the screen (4×2) as shown in Figure 4.

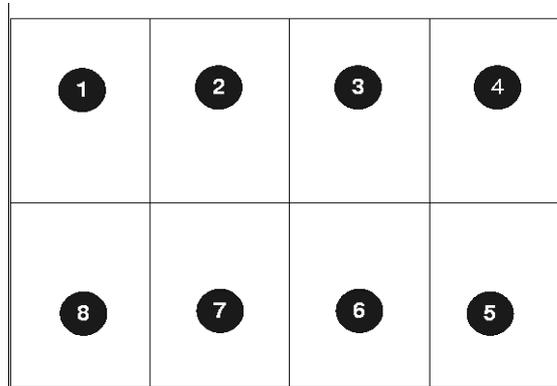


Figure 4: The quantized eye gaze regions on a computer screen

3.1 Pupil and Glint Detection and Tracking

Gaze estimation starts with pupil and glint detection and tracking. For gaze estimation, we continue using the IR illuminator as shown in Figure 2. To produce the desired pupil effects, the two rings are turned on and off alternately via the video decoder we developed to produce the so-called bright and dark pupil effect as shown in Figure 5 (top) and (down).

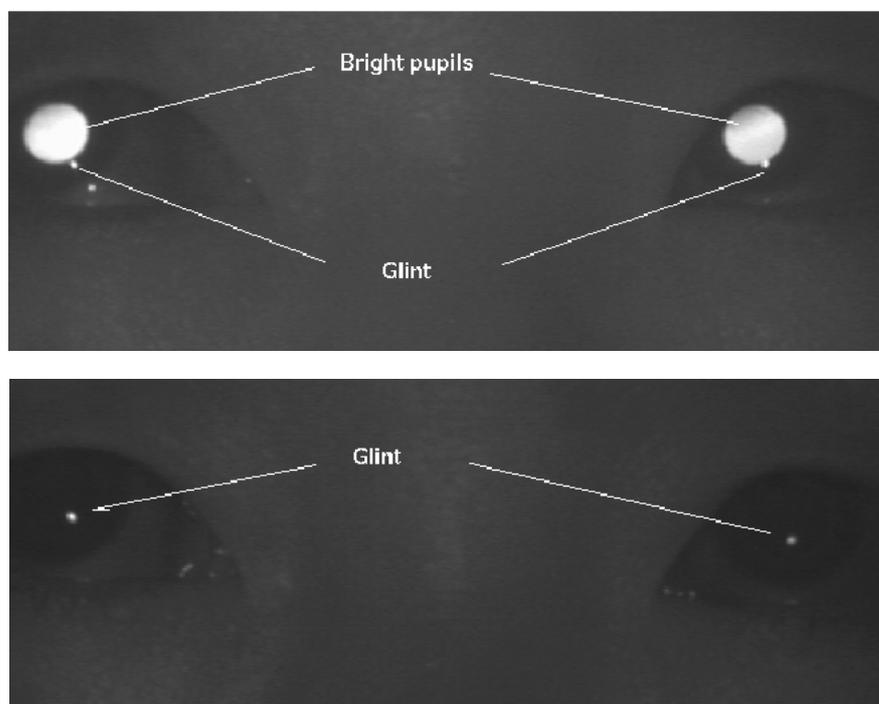


Figure 5: Bright (top) and dark (down) pupils images with glints

Note that glint (the small brightest spot) appears on both images. Given a bright pupil image, the pupil detection and tracking technique described in section 2 can be directly applied for pupil detection and tracking. The location of a pupil at each frame is characterized by its centroid. Algorithm-wise, glint can be detected much more easily from the dark image since both glint and pupil appear equally bright and sometimes overlap on the bright pupil image. On the other hand, in the dark image, the glint is much brighter than the rest of the eye image, which makes glint detection and tracking much easier. The pupil detection and tracking technique can be used to detect and track glint from the dark images.

3.2 Local Gaze Calibration

Given the detected glint and pupil, a mapping function is often used to map the pupil-glint vector to gaze (screen coordinates). The mapping function is often determined via a calibration procedure. The calibration process determines the parameters for the mapping function given a set of pupil-glint vectors and the corresponding screen coordinates (gazes). The conventional approach for gaze calibration suffers from two shortcomings. First, most of the mapping is assumed to be an analytical function of either linear or second order polynomial, which may not be reasonable due to perspective projection and the spherical surface of the eye. Second, only coordinate displacements between the pupil center and glint position are used for gaze estimation, which makes the calibration face orientation dependent. Another calibration is needed if the head has moved since last calibration, even for minor head movement. In practice, it is difficult to keep the head still (unless a support device like a chin rest is used) and the existing gaze tracking methods will produce an incorrect result if the head moves, even slightly.

3.3 Face Pose by Pupil Properties

In our pupil tracking experiments, we had an interesting observation that the pupil appearances vary with different poses. Our study shows that there exists a direct correlation between 3D face pose and the properties such as pupils' size, inter-pupil distance, pupils' shape and pupil ellipse's orientation. It is apparent from these images that

- (1) the inter-pupil distance decreases as the face rotates away from the frontal orientation.
- (2) the ratio between the average intensity of two pupils either increases to over one or decreases to less than one as face rotates away or rotates up/down.
- (3) the shapes of two pupils become more elliptical as the face rotates away or rotates up/down.
- (4) the sizes of the pupils also decrease as the face rotates away or rotates up/down.
- (5) the orientation of the pupil ellipse will change as the face rotates around the camera optical axis.

Based on the above observations, we developed a face pose classification algorithm by exploiting the relationships between face orientation and these pupil parameters.

Details on the face pose estimation based on pupil parameters may be found in Ji & Yang (2001).

3.4 Parameters for Gaze Calibration

In order to incorporate the face pose and the local gaze information into the gaze tracker, the factors accounting for the head movements and those affecting the local gaze should be combined to produce the final gaze. Hence, six factors are chosen for the gaze calibration to get the mapping function: Δx , Δy , r , θ , g_x , and g_y . Δx and Δy are the pupil-glint displacement. r is the ratio of the major to minor axes of the ellipse that fits to the pupil. θ is the pupil ellipse orientation and g_x , and g_y are the glint image coordinates. The choice of these factors is based on the following rationale. Δx and Δy account for the relative movement between the glint and the pupil, representing the local gaze. The magnitude of the glint-pupil vector can also relate to the distance of the subject to the camera. r is used to account for out-of-plane face rotation. The ratio should be close to one when the face is frontal. The ratio becomes larger or less than 1 when the face turns either up/down or left/right. Angle θ is used to account for in-plane face rotation around the camera optical axis. Finally, (g_x, g_y) is used to account for the in-plane head translation.

The use of these parameters accounts for both head and pupil movement since their movements will introduce corresponding changes to these parameters. This effectively reduces the head movement influence. Furthermore, the input parameters are chosen such that they remain relatively invariant for different people. For example, these parameters are independent of the size of the pupils, which often vary among people. The determined gaze mapping function, therefore, will be able to generalize. This effectively eliminates the need to re-calibrate for another person.

4. Gaze Calibration via Generalized Regression Neural Networks (GRNN)

Given the six parameters affecting gaze, we now need to determine the mapping function that maps the parameters to the actual gaze. The reason to use neural networks (NN) to determine the mapping function is because of the difficulty in analytically deriving the mapping function that relates pupil and glint parameters to gaze under different face poses and for different persons. Given sufficient pupil and glint parameters, we believe there exists a unique function that relates gaze to different pupil and glint parameters.

Introduced in 1991 by Specht (Specht, 1991) as a generalization of both radial basis function networks (RBFNs) and probabilistic neural networks (PNNs), GRNNs have been successfully used in function approximation applications. GRNNs are memory-based feed forward networks based on the estimation of probability density functions. GRNNs feature fast training times, can model non-linear functions, and have been shown to perform well in noisy environments given enough data. Our experiments with different types of NN also reveal superior performance of GRNN over the conventional feed forward back propagation NN.

The GRNN topology consists of 4 layers: the input layer, the hidden layer, the summation layer and the output layer. The input layer has six inputs, representing the

six parameters while the output layer has one node. The number of hidden nodes is equal to the number of training samples, with one hidden node added for each set of the training sample. The number of nodes in the summation layer is equal to the number of output nodes plus 1. Figure 6 shows the GRNN architecture we use.

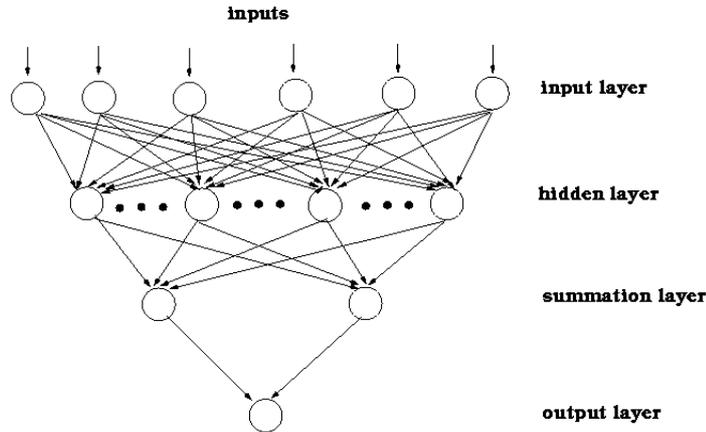


Figure 6: GRNN architecture used for gaze calibration

Due to a significant difference in horizontal and vertical spatial gaze resolution, two identical GRNN networks were constructed, with output node representing the horizontal and vertical gaze coordinates S_x and S_y and respectively.

The parameters to use for the input layer must vary with different face distances and orientations to the camera. Specifically, the input vector to the GRNN is

$$g = \left[\Delta x \quad \Delta y \quad r \quad \theta \quad g_x \quad g_y \right]$$

Before supplying to the GRNN, the input vector is normalized appropriately. The normalization ensures that all input features are in the same range. A large amount of training data under different head positions is collected to train the GRNN. During the training data acquisition, the user is asked to fixate his/her gaze on each gaze region. On each fixation, 10 sets of input parameters are collected so that outliers can be identified subsequently. Furthermore, to collect representative data, we use one subject from each race including an Asian subject and a Caucasian subject. In the future, we will extend the training to additional races. The subjects' ages range from 25 to 65. The acquired training data, after appropriate preprocessing (e.g., non-linear filtering to remove outliers) and normalization, is then used to train the NN to obtain the weights of the GRNN. GRNNs are trained using a one-pass learning algorithm and the training is therefore very fast.

4.1 Gaze Mapping and Classification

After training, given an input vector, the GRNN can then classify it into one of the 8 screen regions. Our experiment, as summarized in Table 1, shows that an average of gaze classification accuracy of 85% was achieved for 480 testing data not included in the training data. Further analysis of this result shows significant misclassifications occur between neighboring gaze regions. For example, about 18% of the gazes in

region 1 are misclassified to gaze region 2 while about 24% gazes for region 3 are misclassified as gaze region 4. We therefore conclude misclassification almost exclusively occurs among neighboring gaze regions.

Table 1: Gaze classification results for the one-level GRNN classifier. An average of gaze classification accuracy of 85% was achieved for 480 testing data not included in the training data for the one level gaze classifier.

ground truth (target #)	estimated result (mapping target #)								Correctness rate (%)
	1	2	3	4	5	6	7	8	
1	49	11	0	0	0	0	0	0	82
2	0	52	8	0	0	0	0	0	87
3	0	0	46	14	0	0	0	0	77
4	0	0	0	59	1	0	0	0	98
5	0	0	0	0	60	0	0	0	100
6	0	0	0	6	8	46	0	0	77
7	0	0	2	0	0	5	53	0	88
8	4	0	0	0	0	0	6	50	84

4.2 Hierarchical Gaze Classifier

To reduce misclassification among neighboring gaze classes, we designed a hierarchical classifier to perform additional classification. The idea is to focus on the gaze regions that tend to get misclassified and perform reclassification for these regions. Therefore, we designed a classifier for each gaze region to perform the neighboring classification again. According to the regions defined in Figure 4, we first identify the neighbors for each gaze region and then only use the training data for the gaze region and its neighbors to train the classifier. Specifically, each gaze region and its neighbors are identified as follows.

- (1) Region 1: neighbors: 2,8
- (2) Region 2: neighbors: 1,3,7
- (3) Region 3: neighbors: 2,4,6
- (4) Region 4: neighbors: 3,5
- (5) Region 5: neighbors: 4,6
- (6) Region 6: neighbors: 3,5,7
- (7) Region 7: neighbors: 2,6,8
- (8) Region 8: neighbors: 1,7

These sub-classifiers are then trained using the training data consisting of the neighbors' regions only. The sub-classifiers are subsequently combined with the whole-classifier to construct a hierarchical gaze classifier as shown as Figure 7.

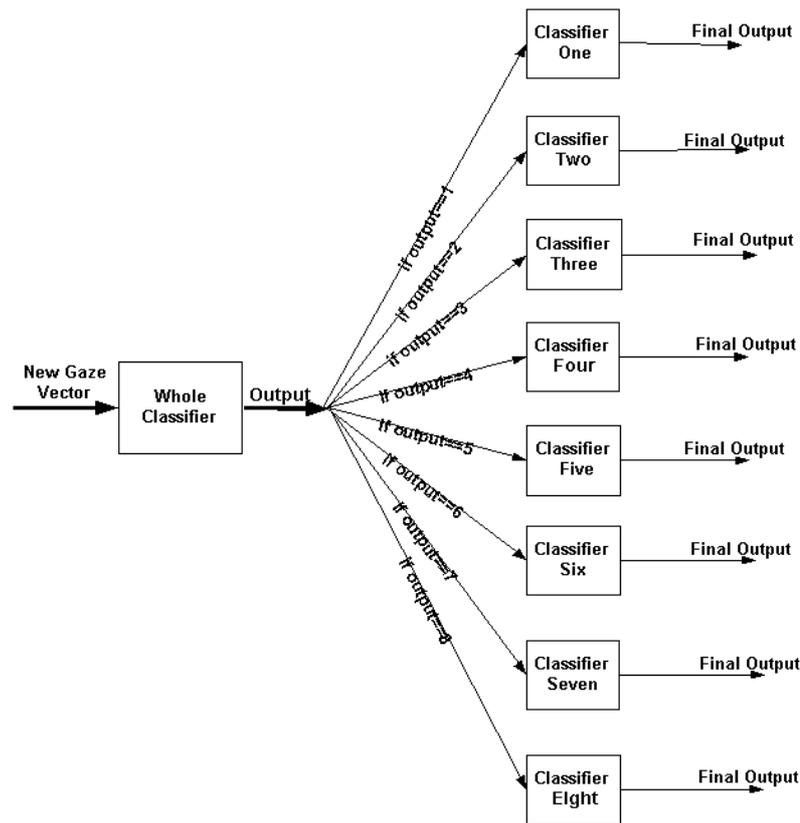


Figure 7: The structure of the hierarchical gaze classifier

Given an input vector, the hierarchical gaze classifier works as follows: first, the whole classifier classifies the input vector into one of the eight gaze regions; then, according to the classified region, the corresponding sub-classifier is activated to reclassify the input vector to the gaze regions covered by the sub-classifier. The result obtained from the sub-classifier will be considered as the final classified result. Our expectation is that the final classification results should improve or remain the same at least and will not get worse. Our experiments prove this.

5. Experimental Results and Analysis

To validate the performance of our gaze tracker, we perform a series of experiments that involves the use of gaze to interactively determine what to display on the screen.

The first experiment involves visual evaluation of our eye tracking system. A laser pointer is used to point at the different regions of the computer screen. As expected, the user gaze is able to accurately follow the movement of the laser pointer which moves randomly from one gaze region to another gaze region, even under natural head movement. A video demo of this experiment is available at <http://www.ecse.rpi.edu/~cvrl>.

To quantitatively characterize the accuracy of our system, the second experiment studies the performance of our system under different face orientations and distances to the cameras and with different subjects. Table 2 summarizes the classification results. Compared with Table 1, which was produced based on the same data, we can see that the hierarchical gaze classifier can achieve an average of around 95%

accuracy for a different subject, which improves the accuracy by around 10% over the one level gaze classifier method. Specifically, the misclassification rate between neighbors 1 and 2 was reduced from 18% to about 8% while the misclassification rate between gaze regions 3 and 4 was reduced to about 5% from the previous 24%. The classification errors for other gaze regions have also improved or remained unchanged. The hierarchical classification therefore meets our expectation.

Table 2: An average of gaze classification results (95% accuracy) was achieved for 480 testing data not included in the training data for the hierarchical gaze classifier.

ground truth (target #)	estimated result (mapping target #)								Correctness rate (%)
	1	2	3	4	5	6	7	8	
1	55	5	0	0	0	0	0	0	92
2	0	58	2	0	0	0	0	0	97
3	0	0	57	3	0	0	0	0	95
4	0	0	0	59	1	0	0	0	98
5	0	0	0	0	60	0	0	0	100
6	0	0	1	5	5	49	0	0	82
7	0	0	2	0	0	5	53	0	88
8	3	0	0	0	0	0	2	55	92

Our study, however, shows that our system has some difficulty with older people, especially for those who suffer from some vision problem such as far-sightedness or near-sightedness.

Our experiments show that our system, working in near real time (20 Hz) with an image resolution of 640×480 on a Pentium 3, allows about 6 inches left/right and up/down head translational movement and allows ± 20 degrees left/right head rotation as well as ± 15 degrees up/down rotation. The distance to the camera ranges from 3.5 feet to 5 feet. The spatial gaze resolution is about 5 degrees horizontally and 8 degrees vertically, which corresponds to about 4 inches horizontally and 5 inches vertically at a distance about 4 feet away from the screen.

Finally, we apply our gaze tracker for natural user computer interaction. For this experiment, the screen is divided into 2×4 regions, with each region labeled with a word such as *water* or *phone* to represent the user's intention or needs. Figure 8 shows the regions of the computer screen with labeled words. During the experiment, the user sits in front of the computer naturally and gazes at different region of the screen. If the user's gaze fixation at a region exceeds a predefined threshold, an audio sound is uttered by the speaker to express the user intention as determined by the labeled word of the gazed region. For example, if the user gazes at the region that contains the word **water** for some time, then the PC speaker will read "Please bring me a cup of water." This experiment repeats until the user decides to quit. For a real-time demonstration of the gaze tracking software, please refer to <http://www.ecse.rpi.edu/~cvrl>.

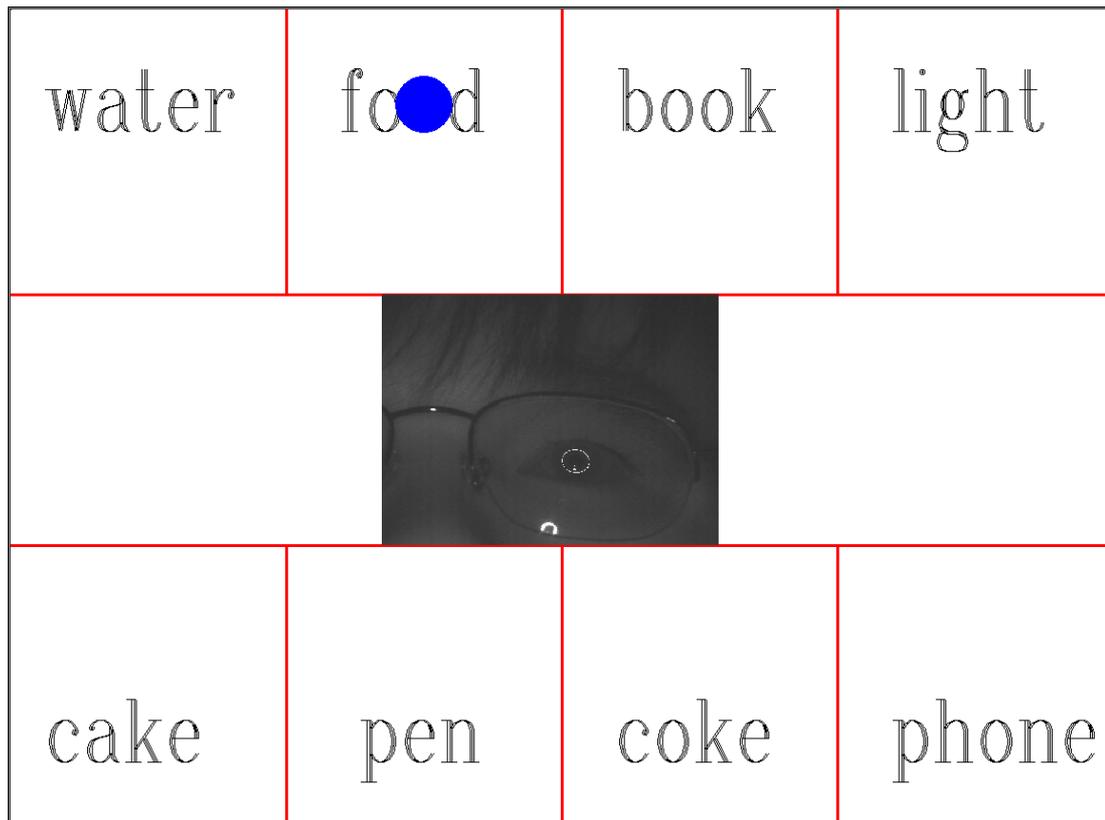


Figure 8: The user computer interaction screen, which is divided into 8 regions and each corresponds a particular need of the user.

6. Conclusion

In this paper, we present a new approach for gaze tracking. Compared with the existing gaze tracking methods, our method, though at a lower spatial gaze resolution (about 5 degrees), has the following benefits: no calibration is necessary, it allows natural head movement, and it is completely non-intrusive and unobtrusive while still producing relatively robust and accurate gaze tracking. The improvement results from using a new gaze calibration procedure based on GRNN. With GRNN, we do not need to assume an analytical gaze mapping function, therefore we can account for head movements in the mapping. The use of hierarchical classification schemes further improves the gaze classification accuracy.

While our gaze tracker may not be as accurate as some commercial gaze trackers, it achieves sufficient accuracy even under large head movements and, more importantly, it is calibration-free. It has significantly relaxed the constraints imposed by most existing commercial eye trackers. We believe, after further improvement, our system will find many applications including smart graphics, human computer interaction, non-verbal communications via gaze, and assistance of people with disabilities.

7. Acknowledgements

The research described in this report is supported by a grant from AFOSR.

8. References

- Bour, L. (1997). *Dmi-search scleral coil*. Tech. Rep.: H2-214, Dept. of Neurology, Clinical Neurophysiology, Academic Medical Center, University of Amsterdam, Amsterdam, Netherlands.
- Ebisawa, Y. (1995). *Unconstrained pupil detection technique using two light sources and the image difference method*. Visualization and Intelligent Design in Engineering and Architecture, pages 79-89, 1995.
- Ebisawa, Y. (1998). *Improved video-based eye-gaze detection method*. IEEE Transactions on Instrumentation and Measurement, 47(2): 948-955, 1998.
- Gee, A. & Cipoll, R. (1994). *Determining the gaze of faces in images*. Image and Vision Computing, 30:639-647, 1994.
- Gips, J.; Olivieri, P. & Tecce, J. (1993). *Direct control of the computer through electrodes placed around the eyes*. Human-Computer Interaction: Applications and Case Studies, pages 630-635, 1993.
- Hutchinson, T.E. (1988). *Eye movement detection with improved calibration and speed*. United States Patent [19], (4,950,069), 1988.
- Hutchinson, T.E.; White, K.; Worthy, J.R.; Martin, N.; Kelly, C.; Lisa, R. & Frey, A. (1989). *Human-computer interaction using eye-gaze input*. IEEE Transactions on systems, man, and cybernetics, 19(6):1527-1533, 1989.
- Ji, Q. (2002). *3D face pose estimation and tracking from a monocular camera*. Image and Vision Computing, 20:499-511, 2002.
- Ji, Q. & Yang, X. (2001). *Real time visual cues extraction for monitoring driver vigilance*. In: ICVS 2001: Second International Workshop on Computer Vision Systems, Vancouver, Canada, 2001.
- Ji, Q. & Yang, X. (2002). *Real time 3D face pose discrimination based on active IR illumination*. International Conference on Pattern Recognition, 2002.
- Koons, D. & Flickner, M. IBM blue eyes project. <http://www.almaden.ibm.com/cs/blueeyes>.
- Morimoto, C.H.; Koons, N.; Amir, A. & Flickner, M. (1999). *Frame-rate pupil detector and gaze tracker*. IEEE ICCV'99 FRAME-RATE Workshop, 1999.
- Motwani, M.C. & Ji, Q. (2001). *3D Face Pose Discrimination Using Wavelets*. IEEE International Conference on Image Processing (ICIP'2001), Thessaloniki, Greece, October 7-10, 2001.
- Ohno, T.; Mukawa, N. & Yoshikawa, A. (2002). *Freegaze: A gaze tracking system for everyday gaze interaction*. Eye Tracking Research and Applications Symposium, 25-27 March, New Orleans, LA, USA, 2002.

- Ritter, H. & Rae, R. (1998). *Recognition of human head orientation based on artificial neural networks*. IEEE Transactions on Neural Networks, 9(2):257-265, 1998.
- Specht, D.F. (1991). *A general regression neural network*. IEEE Transactions on Neural Networks, 2:568-576, 1991.
- Zhu, Z; Fujimura, K. & Ji, Q. (2002). *Real-time eye detection and tracking under various light conditions*. Eye Tracking Research and Applications Symposium, 25-27 March, New Orleans, LA, USA, 2002.