Free head motion eye gaze tracking using a single camera and multiple light sources

Flávio Luiz Coutinho and Carlos Hitoshi Morimoto
Departamento de Ciência da Computação
Instituto de Matemática e Estatística – IME-USP
Rua do Matão 1010, São Paulo, SP
{flc, hitoshi}@ime.usp.br

Abstract

One of the main limitations of current remote eye gaze tracking (REGT) techniques is that the user's head must remain within a very limited area in front of the monitor screen. In this paper we present a free head motion REGT technique. By projecting a known rectangular pattern of lights, the technique estimates the gaze position relative to this rectangle using an invariant property of projective geometry. We carry extensive analysis of similar methods using an eye model to compare their accuracy. Based on these results, we propose a new estimation procedure that compensates the angular difference between the eye visual axis and optical axis. We have developed a real time (30 fps) prototype using a single camera and 5 light sources to generate the light pattern. Experimental results shows that the accuracy of the system is about 1^o of visual angle.

1 Introduction

Eye gaze trackers are devices that estimate the eye gaze direction. Primarily used in laboratories under controlled conditions, several eye gaze enhanced computer interfaces have been suggested in the literature [2] but, despite their great potential, their successes are still limited to a few specific applications.

For interactive applications, image based remote eye gaze trackers (REGT) [1] offer comfort, faster setup, and an accuracy of about 1^o of visual angle. In a recent survey, Morimoto and Mimica [4] identify two main limitations of current REGT technology: their frequent need of calibration and their susceptibility to head motion, often requiring a chin rest or bite bar to maintain the system's accuracy. In the survey, the authors describe a new generation of REGT that minimize these problems.

Some of the new techniques explicitly use depth information from stereo cameras [1, 5]. Avoiding complex hardware setups, Yoo *et al.* [8] introduced a single camera multiple light source technique that uses an invariant property of projective geometry (i.e., the cross ratio). Later, Yoo and Chung [7] further refined the technique introducing several enhancements that considerably increased its accuracy, but that still relies on a very simple eye model.

In this paper, we extend the light pattern projection technique using a more realistic model of the eye, and a new calibration procedure that computes the optimum value of the parameter α used to adjust the position of the glints so that they lie on a virtual plane. The next section describe in more detail the light pattern projection technique. In order to better understand the behavior of the system to different head positions, we conducted extensive simulations of the available techniques using a realistic eye model. The results of these simulations are discussed in Section 3. In Section 4 we introduce a new method to compute the virtual glints that compensates the difference between the eye visual axis to its optical axis, and in Section 5 we describe a real time implementation of this new REGT. In Section 6 we discuss the results of some experiments with the prototype, and Section 7 concludes the paper.

2 Free Head Motion REGT

The light pattern projection method proposed by Yoo *et al.* [8] makes use 5 IR LEDs and a camera. Four of them are fixed at the corners of the computer's screen generating 4 corneal reflections on captured images of the eye. The other LED is placed on the optical axis of the camera and generates bright pupil images which helps the detection of the pupil. It is assumed the existence of a virtual tangent plane to the cornea and the quad (quadrilateral) formed by

[1] They are also known as non-intrusive because they do not require any equipment in direct physical contact with the user, such as electrodes or contact lenses.

XIX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'06)
0-7695-2686-1/06 $20.00 © 2006 IEEE

the 4 glints to be a projection of the screen on this plane. In the same way, the center of the pupil is a projection of the observed screen point on the plane. Since the pupil and the four glints are projections of the gaze point and the four LEDs, respectively, their cross ratio, that is invariant in projective space, can be used to estimate the gaze point.

This method was later extend by Yoo and Chung [7] where the geometry of the previous system were refined. Once the corneal reflections generated by the monitor's LEDs are not really projections on the virtual plane, the authors developed a way to estimate these projections (which were called virtual projections since they are formed on the virtual plane) from the glints. Besides this, the new system makes use of two cameras, mounted on a pan-tilt unit, instead of only one. One of the cameras has a wide field of view and is responsible for locating the user's face and keep the other camera (a narrow field one) always pointing to one of the user's eyes.

Figure 1 shows how the corneal reflection is formed for one of the monitor's LEDs and how it is related to the virtual projection of the same LED. Let L_1 be one of the four LEDs attached to the monitor, O the center of the cornea and C the center of projection of the camera. V_1 and R_1 are, respectively, the virtual projection and corneal reflection of L_1, and R_C is at the same time the reflection and the virtual projection of the LED fixed at the camera's optical axis. When an image of the eye is captured, V_1, R_1 and R_C are projected to the image plane as U_{V1}, U_{R1}, and U_{RC}. They showed that, considering the typical positioning of a user, $d(U_{V1}, U_{RC}) \approx 2d(U_{R1}, U_{RC})$, where $d(X, Y)$ is the distance between the points X and Y. To estimate the virtual projections from the corneal reflections, each vector $\overrightarrow{U_{RC}U_{Ri}}$ is scaled by a factor α, which is close to 2.0, using the following equation:

$$U_{Vi} = \alpha(U_{Ri} - U_{RC}) + U_{RC} \qquad (1)$$

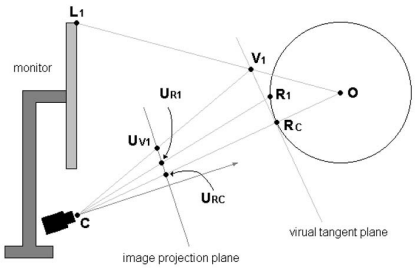

Figure 1. Side view of monitor showing the formation of corneal reflection and virtual projection for one of the monitor's LEDs.

Given a captured image of the eye, with the pupil (U_P) and the five corneal reflections (U_{R1}, U_{R2}, U_{R3}, U_{R4} and U_{RC}) it is possible to calculate the points U_{V1}, U_{V2}, U_{V3}, U_{V4} which are the result of two projective transformations over the points L_1, L_2, L_3 and L_4. The pupil center (P) located on the surface of the cornea can also be considered as the projection of the point of regard (G) on the virtual plane and the point U_P is the result of the application of two projective transformations over G. This way, as mentioned earlier, it is possible to use the cross-ratio to estimate G, which is defined as follows:

$$cr(P1, P2, P3, P4) = \frac{|P_1P_2||P_3P_4|}{|P_1P_3||P_2P_4|} \qquad (2)$$

where $|P_iP_j|$ is the determinant of the matrix M whose first column corresponds to the coordinates of P_i and the second column to the coordinates of P_j. The important property of the cross-ratio is that if the line containing P_1, P_2, P_3 and P_4 are subject to any projective transformation, the ratio remains the same [3].

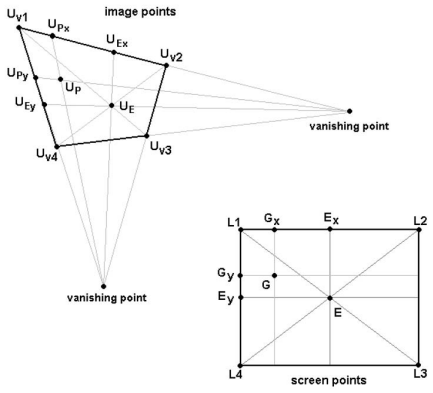

Figure 2. Screen points and corresponding image points used to estimate G by application of the cross-ratio.

Figure 2 shows how to apply the cross-ratio to estimate the gaze point. After calculating the virtual projections U_{V1}, U_{V2}, U_{V3} and U_{V4} it is possible to calculate the point U_E (by the intersection of the diagonals of the quad $U_{V1}U_{V2}U_{V3}V_{U4}$, which corresponds to the projection of the central point of the screen E) and then U_{Ex}, U_{Ey}, U_{Px} and U_{Py}. These points define two lines, each one with four collinear points, which are projections of the segments $\overline{L_1L_2}$ and $\overline{L_1L_4}$. Calculating one cross-ratio for each of these lines, G_x and G_y can also be calculated and therefore, G can be estimated.

Since the pupil is not located on the cornea surface, the α value of 2.0 may not be suitable for the estimation of the virtual projections in practice. To deal with this issue, the

authors of the technique proposed a calibration method for calculating α. They use different α values to estimate each one of the four LEDs virtual projections and each one is calculated by the following equation:

$$\alpha_i = \frac{d(U_P, U_{RC})}{d(U_{Ri}, U_{RC})} \quad (3)$$

The idea of this calibration procedure is that when a user looks at one of the LEDs, the pupil center should match the virtual projection of the corresponding LED. Considering a more realistic eye model (we will consider the Gullstrand's eye model [6]), we expect that when the eye axes difference is considered this calibration will not be accurate, since the pupil center is associated with the optical axis and the real gaze line with the visual axis.

3 Simulation of the light pattern projection methods

In order to observe the behavior of the technique based on the light pattern projection, several simulations were carried out, where the cornea was modeled as a sphere with radius of 0.77 cm, initially positioned at P_0 = (0, 27, 60). The camera was placed at (0, 0, 0) and the monitor's LEDs were positioned at (-18.3, 27.4, 0), (18.3, 27.4, 0), (18.3, 0, 0) and (-18.3, 0, 0). The pupil has been considered to be located at the cornea surface and the vector connecting O to P is defined as the optical axis, with initial value of (0, 0, -1).

To better understand how rotations of the eye were modeled, we will consider the following coordinate system for the eye: cornea at origin, y axis pointing upwards (the same direction of the world's y axis), z pointing in the direction of the optical axis and x defined by the cross product of y and z. When the eye is rotated, to point to a specific direction, a composition of two rotations is applied over the eye's initial condition. The first rotation is made over y and the next over the $R_y(x)$ axis (i.e. the x axis transformed by the first rotation over y). 人眼的视轴和光轴有偏差，把眼球建模成球形时忽略了这种偏差。

To model the eye axes difference, the optical axis is rotated to define the visual axis. We used rotation values of 4.5^o and 3.0^o, resulting in an angular difference of approximately 5.5^o between the axes (which is in agreement with the real angular difference of 4-8^o observed in the human eye [6]). Since it is the visual axis that really points to the observed point, when the eye model is directed to a certain point we apply a rotation so that the visual axis, and not the optic, points towards it.

A simulation consists of translating the eye 10 cm in every direction for all axes in space, generating a total of 7 possible positions (including the initial one). For each position the eye is rotated so that it observes 48 test points placed at the center of each element of a 8×6 grid that covers the entire screen. For each eye position and observed point, we calculate the locations of the corneal reflections, project them along with the pupil to the image plane and calculate the virtual projections. Then we estimated the gaze point and compare it to the real observed point.

Since we are moving the eye in space, we can see if the technique is robust to user head movement. In addition, to observe how other parameters affect the results, several simulations were done using different methods for estimation of the virtual projections of the LEDs and taking or not the eye axes difference into consideration. When this difference is not considered, the visual axis is defined being equal to the optical axis.

Tree methods for estimation of the virtual projections were tested. The first one, consists in the direct use of the corneal reflections, similar to the method described in [8]. In the second method, we estimate each virtual projection using an α value of 2.0, which is the approximate expected value for the correction factor as seen in the previous section. Finally, the third one, is the method proposed by Yoo and Chung [7] in which a calibration procedure is performed to determine one value of α for each virtual projection being calculated.

3.1 Corneal reflections

Results obtained by direct use of the corneal reflections in gaze estimation can be seen in Figures 3 and 4, and column "Reflection" of Table 1. We can note that the results were very poor, considering or not the axes difference of the eye, showing the need to estimate the virtual projections to calculate the gaze point. In the case where the axes difference was considered, it is interesting to note that the set of estimated points is similar to the set of estimated points obtained in the other test, with the difference that they appear to be translated.

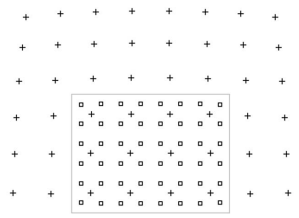

Figure 3. Estimated (crosses) and real (squares) points for the cornea located at P_0. In this test direct use of the corneal reflections was made to estimate the gaze points. The eye axes difference was not considered.

XIX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'06)
0-7695-2686-1/06 $20.00 © 2006 IEEE

	Reflection		α = 2.0		Yoo method		New method	
Position	**optical**	**visual**	**optical**	**visual**	**optical**	**visual**	**4 points**	**9 points**
initial	16.1 ± 5.3	22.5 ± 5.7	0.3 ± 0.2	6.0 ± 0.4	0.7 ± 0.3	4.2 ± 1.0	1.0 ± 0.3	0.5 ± 0.3
x $(-)$	16.1 ± 5.4	23.2 ± 6.8	0.3 ± 0.2	6.1 ± 0.3	0.7 ± 0.3	4.1 ± 0.9	1.1 ± 0.3	0.6 ± 0.3
x (+)	16.1 ± 5.3	22.1 ± 4.9	0.3 ± 0.2	6.2 ± 0.6	0.7 ± 0.3	4.5 ± 1.2	1.1 ± 0.4	0.6 ± 0.3
y $(-)$	17.3 ± 6.1	23.8 ± 6.7	0.3 ± 0.2	5.9 ± 0.3	0.8 ± 0.3	4.1 ± 1.0	1.0 ± 0.3	0.4 ± 0.2
y (+)	15.2 ± 4.8	21.9 ± 4.9	0.3 ± 0.2	6.3 ± 0.5	0.7 ± 0.3	4.5 ± 1.1	1.2 ± 0.3	0.7 ± 0.3
z $(-)$	15.7 ± 4.9	20.7 ± 5.0	0.4 ± 0.3	5.2 ± 0.4	0.8 ± 0.4	3.5 ± 1.0	1.0 ± 0.5	1.0 ± 0.4
z (+)	16.5 ± 5.7	24.2 ± 6.2	0.2 ± 0.1	6.9 ± 0.3	0.7 ± 0.3	5.0 ± 1.1	1.6 ± 0.2	1.0 ± 0.2

Table 1. Average error (in cm) obtained for several executions of the simulation. Four different methods to estimate the virtual projections were tested. For the first three methods a simulation was made without considering the axes difference (columns marked with "optical") and another one was made considering it ("visual"). For the last method, one simulation was made using 4 calibration points ("4 points") and another with 9 calibration points ("9 points"). (+) and $(-)$ indicate, respectively, positive and negative translations of 10 cm along the specified axis.

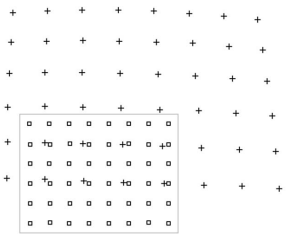

Figure 4. Estimated (crosses) and real (squares) points for the cornea at P_0. In this test direct use of the corneal reflections was made to estimate the gaze points. The eye axes difference was considered for this test.

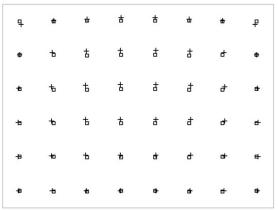

Figure 5. Estimated (crosses) and real (squares) points for the cornea at P_0. All virtual projections were estimated using α = 2.0. The eye axes difference was not considered.

3.2 Using a constant $\alpha = 2.0$

Using an α value of 2.0 to estimate all the virtual projections produced much more accurate results as can be seen in Figure 5 and column "α = 2.0: optical" of Table 1 when we do not consider the difference of the eye's axes. However, when the difference is considered, precision in the estimation was not as good (see Figure 6 and column "α = 2.0: visual" of Table 1). Nevertheless, we can observe that the offset between each estimated point and the corresponding test point is approximately uniform suggesting that the axes difference could be corrected by an offset vector applied to each estimated point.

3.3 Yoo's calibration method

When using the calibration procedure proposed by Yoo and Chung [7] to compute an α value for each virtual projection being calculated, the results were the following: when the axes difference were not considered good results were achieved (see Figure 7 and column "Yoo method: optical" of Table 1). However, when the axes difference were taken into consideration, the accuracy of the results dropped considerably (see Figure 8 and column "Yoo method: visual" of Table 1). Besides this, we could no longer observe the constant offset between the pairs of estimated and real points. This observation indicates that the calibration method does not compensate for eye axes difference since the results are not accurate and the offset constancy are not preserved, showing that this method for calculation of virtual projections should not work very well in practice.

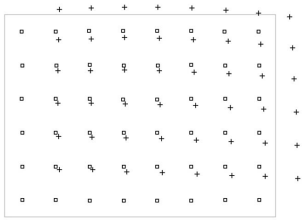

Figure 6. Estimated (crosses) and real (squares) points for the cornea at P_0. All virtual projections were estimated using α = 2.0. The eye axes difference was considered for this test.

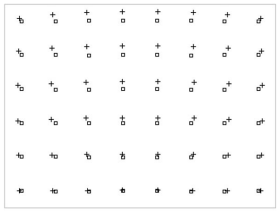

Figure 7. Estimated (crosses) and real (squares) points for the cornea at P_0. Virtual projections were estimated using the method proposed by Yoo and Chung. The eye axes difference was not considered.

4 New method for virtual projection estimation

The results from the simulations from the previous section lead us to the following conclusions:

- it is necessary to compute the virtual projections of the LEDs when estimating the gaze point.
- it is possible to compute the virtual projections using a single α value as the scale factor.
- the axes difference of the eye must be considered and can be compensated by adding an offset vector to the estimated points. However this solution has some sensibility to eye movements along the z axis.

Based on these observations, and considering the fact that the calibration method proposed by Yoo and Chung did not produced good results, a new method to compute the virtual projections were developed. The new method uses just one α for all virtual projections obtained by a different calibration procedure which also calculates an offset vector that is used to compensate the difference in the eye axes.

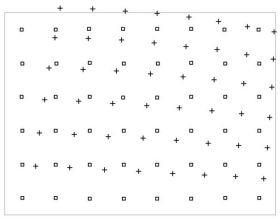

Figure 8. Estimated (crosses) and real (squares) points for the cornea at P_0. Virtual projections were estimated using the method proposed by Yoo and Chung. The eye axes difference was considered for this test.

The idea is to calculate an α value that generates a set of estimated points that can also be described as a translation of the real observed points (in this case, the calibration points). Using n calibration points, consider

$$C = \{C_i | i \in [1..n]\}$$

as the set of such points,

$$E_\alpha = \{E_{\alpha i} | E_{\alpha i} = T(C_i, \alpha), i \in [1..n]\}$$

the set of the estimated points (where $T(C_i, \alpha)$ corresponds to the application of the gaze tracking algorithm for calibration data relative to point C_i and the given α value),

$$D_\alpha = \{\overrightarrow{d_{\alpha i}} | \overrightarrow{d_{\alpha i}} = (C_i - E_{\alpha i}), i \in [1..n]\}$$

the set of offset vectors and $\overrightarrow{m_\alpha}$ the average offset vector. In the ideal case, for the best value of α it would be expected that each element of D_α to be equal to $\overrightarrow{m_\alpha}$. Since there is some variation of the offset vectors, the best value of α is the one that generates a D_α set with elements as uniform as possible. In other words, we want an α value that minimizes the following sum:

$$sum(\alpha) = \sum_{i=1}^{n} |\overrightarrow{d_{\alpha i}} - \overrightarrow{m_\alpha}| \quad (4)$$

The plot of sum(α) versus α shows a curve that starts with a negative slope, reaches a minimum value for the sum, and then starts to grow again. Placing the eye at different positions, the general form of the plot remains the same, so we can use a bisection method for the computation of the optimum α value that generates the minimum sum. Once α is calculated, the vector $\overrightarrow{m_\alpha}$ is taken as the average offset to be used to correct estimation errors due to the axes difference of the eye.

To verify the behaviour of the new method, we simulated it under the same conditions used for the other methods presented in the previous section. Two tests were made, one using 4 calibration points that correspond to the positions of the LEDs on the screen, and the other using 9 calibration points corresponding to the center of each element obtained by dividing the screen as a regular 3×3 grid. Results of the first test can be seen in Figure 9 and column "New method: 4 points" of Table 1. Results of the second test are shown in Figure 10 and column "New method: 9 points" of Table 1. Notice that the new method works well and achieves better performance using 9 calibration points because they offer a more detailed sample of the screen. Notice also that there is some loss of precision for translations over the z axis. Even though, the tolerance for translations in all directions is better than those observed in the experiments presented in Morimoto and Mimica [4], which used similar experimental conditions to analyse the traditional gaze tracking technique based on the pupil and corneal reflection tracking. Table 2 shows the comparison of head movement tolerance for each technique. In Figure 11 it is possible to see how the estimation errors are affected by large translations using the new method for estimation of the virtual projections (with 9 calibration points). Errors for translations in the z axis grows almost twice as fast than for translations in the x and y axes. For these two axes, even for translations up to 20 cm, we have an average error about 1 cm (that roughly corresponds to 1^o of visual angle).

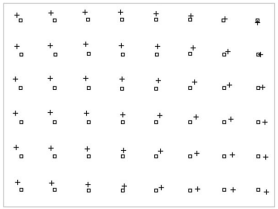

Figure 9. Estimated (crosses) and real (squares) points for the cornea at P_0. Virtual projections were estimated using the new proposed method with 4 calibration points.

5 Implementation issues

Our implementation of the gaze tracker uses a Athlon 1.4 Ghz CPU with 512 Mb of RAM, an Osprey 100 video capture card, a 17 inches LCD monitor and a camera with two sets of IR LEDs (one on the optical axis of the camera, and the other on the monitor's corners as can be seen in Figure 12). An external circuit synchronizes the activation of each set of LEDs with the scan of even/odd lines of a frame.

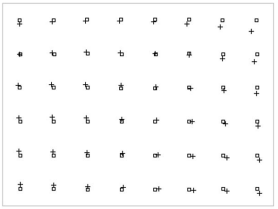

Figure 10. Estimated (crosses) and real (squares) points for the cornea at P_0. Virtual projections were estimated using the new proposed method with 9 calibration points.

Position	Average error for each technique		
	traditional	cr (4 points)	cr (9 points)
initial	0.80 cm	1.0 cm	0.48 cm
x $(-)$	0.99 cm	1.13 cm	0.55 cm
y $(-)$	2.17 cm	0.95 cm	0.39 cm
z $(+)$	4.05 cm	1.63 cm	0.94 cm

Table 2. Comparison of average error obtained by traditional gaze tracking technique and technique developed by Yoo *et at*. using the new proposed method for estimation of virtual projections. (+) and $(-)$ indicate, respectively, positive and negative translations of 10 cm along the specified axis.

The software was developed on a Linux platform, and basically executes the following steps, at a rate of 30 frames per second: image acquisition, image processing and gaze estimation.

Image acquisition uses the video4linux API (V4L) which offers a standard interface for a large number of capture devices. A lot of the image processing was facilitated by the use of the OpenCV library. In the image processing step, the captured image is first deinterlaced resulting in two images of the eye: one with bright pupil and a corneal reflection generated by the LEDs of the camera; the other with dark pupil and four corneal reflections corresponding to the LEDs placed on the corners of the screen.

Rough estimation of the pupil location is then performed. Reduced versions of the deinterlaced images are used, and the dark pupil image is subtracted from the bright one. A threshold is then applied to the resulting difference image and the largest blob (which cannot be excessively long in width or height) is taken as being the pupil. The approximate location of the pupil center is given by its center of mass, and is used to define a region of interest (ROI) in the full sized deinterlaced images for further processing.

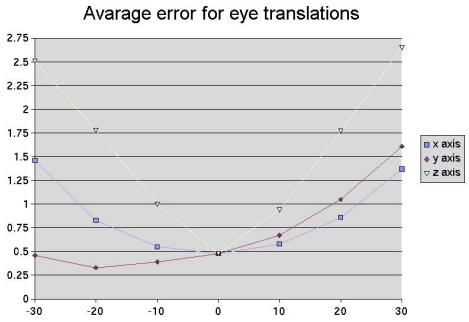

Figure 11. Average error (in cm) obtained by simulations, using the new proposed method for estimation of the virtual projections, for translations of the eye up to 30 cm on both directions of axes x, y and z. In this simulation 9 calibration points were used.

Figure 12. Overview of the gaze tracker's hardware. In the picture we can see the monitor with the IR LEDs attached on it and the camera with a set of IR LEDs on its optical axis.

Glint (corneal reflection) detection is performed next. Since glints appear as small bright regions in the images, a threshold is applied over the full sized deinterlaced images followed by morphological operations and labelling of each resulting connected component. Big components and those located too far from the pupil center are discarded, and the resulting ones become our glint candidates. Since the implemented technique for gaze tracking requires a fixed number of glints to be tracked (4 glints in the dark pupil image and 1 glint in the bright one), the excess or lack of glints are conditions that we need to solve. Unfortunately, when glints are missing we are currently not able to recover. When too many glints are detected in the bright pupil image we simply take the biggest one among the left candidates. For the dark pupil image we need to choose the four correct glints. To do this we take all possible combinations of four glints and choose the combination which forms a quad that is closest to a rectangle.

Having detected all expected glints, precise estimation of the pupil center is done in the full sized images. It is similar to the initial estimation but, instead of calculating the center of mass of the blob that corresponds to the pupil, we take its contour and remove its portions which overlap with the detected glints. The resulting contour is then used to fit an ellipse that best matches the pupil and the ellipse center is taken as the pupil center.

Finally, using all detected feature points, the point of regard is estimated as already discussed. For comparison purpose, both Yoo's method and our proposed method (with 9 calibration points) for estimation of the virtual projections were implemented .

6 Experimental results

In this section we discuss the results of our real time implementation. Five users participated in our experiments. The test procedure consisted in each user looking at 48 points (the centers of each element of a 8×6 grid covering the screen) and comparing the estimated gaze point with the real point. To observe the tracker's tolerance to user head motion each user executed the test procedure, after initial calibration, 3 times and after each execution they were asked to move a little in order to change their positions in space (the camera was also moved to keep the eye within its narrow field of view). The results are shown in Table 3 (first test trial) and Table 4 (other trials) where we can observe a small deterioration of the precision after the execution of the first trial. The average error for all users in all executions of the test was 2.48 cm (about 2.4^o) with standard deviation of 1.12 cm when using Yoo and Chung's method for estimation of the virtual projections. For our method, the average gaze estimation error was 0.95 cm (about 0.91^o) with standard deviation of 0.7 cm. In Figure 13 we can see the estimated points for all tests of all users according to the virtual projection estimation method used. We can observe that in our method, the estimated gaze points are more concentrated around the real observed points. One of the users who participated in the tests wear glasses and another contact-lenses. Despite the fact that glint detection was more difficult for these users, we did not observe any influence of the corrective lenses in the estimation results.

7 Conclusion

In this paper we have presented an extension of the light pattern projection technique introduced by Yoo and Chung [7]. We use a more accurate eye model to correct for the deviation of the eye visual axis from its optical axis, and also

1st test	Yoo method		New method	
User	Error	Std dev	Error	Std dev
user 1*	1.51 cm	0.69 cm	1.23 cm	1.39 cm
user 2	2.51 cm	0.82 cm	0.73 cm	0.29 cm
user 3**	1.75 cm	0.48 cm	0.73 cm	0.39 cm
user 4	3.15 cm	1.13 cm	0.88 cm	0.48 cm
user 5	2.98 cm	0.86 cm	0.78 cm	0.43 cm

Table 3. Gaze estimation results for each user on their first execution of the test, comparing Yoo and Chung method for virtual projections estimation with our proposed method. Users marked with (*) e (**) were wearing, respectively, contact lenses and glasses.

other tests	Yoo method		New method	
User	Error	Std dev	Error	Std dev
user 1*	1.98 cm	0.96 cm	1.15 cm	0.96 cm
user 2	2.53 cm	1.03 cm	0.98 cm	0.62 cm
user 3**	1.8 cm	0.59 cm	0.9 cm	0.55 cm
user 4	2.95 cm	1.15 cm	0.98 cm	0.62 cm
user 5	3.36 cm	1.11 cm	0.93 cm	0.51 cm

Table 4. Gaze estimation results for each user on the other executions of the test, comparing Yoo and Chung method for virtual projections estimation with our proposed method. Users marked with (*) e (**) were wearing, respectively, contact lenses and glasses.

a new calibration procedure to estimate the optimal value of the parameter α, used to adjust the position of the glint to the virtual plane assumption. An extensive analysis of Yoo's method were conducted in order to predict and compare the behavior of the their system and ours. Experimental results obtained using a real time implementation of this new light pattern projection based REGT demonstrates that the accuracy of the system is about 1^o of visual angle and is quite robust to head motion, as predicted by our simulation results.

We are currently developing a method to estimate the distance of the eye to the monitor based on the light pattern alone, to make the system more robust to depth variations. Different light patterns will also be tested to facilitate the detection of the glints and to estimate gaze even when some of the glints are missing.

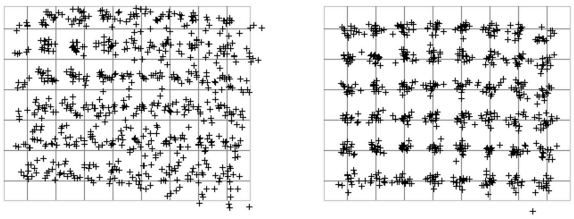

Figure 13. Estimation results for all trials executed by users. Crosses represents the estimated points and line intersections the real observed points. On the left we have the results using Yoo and Chung method for virtual projections estimation. On the right we have the results using our proposed method.

References

[1] D. Beymer and M. Flickner. Eye gaze tracking using an active stereo head. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, volume II, pages 451–458, Madison, WI, June 2003.

[2] A. T. Duchowski. A breadth-first survey of eye tracking applications. *Behavioral Research Methods, Instruments, and Computers*, pages 1–16, 2002.

[3] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000.

[4] C. H. Morimoto and M. R. M. Mimica. Eye gaze tracking techniques for interactive applications. *Comput. Vis. Image Underst.*, 98(1):4–24, 2005.

[5] S. W. Shih and J. Liu. A novel approach to 3d gaze tracking using stereo cameras. *IEEE Transactions on systems, man, and cybernetics - PART B*, 34:234–245, Feb 2004.

[6] G. Wyszecki and W. S. Stiles. *Color Science: Concepts and Methods, Quantitative Data and Formulae*. John Wiley & Sons, 1982.

[7] D. H. Yoo and M. J. Chung. A novel non-intrusive eye gaze estimation using cross-ratio under large head motion. *Comput. Vis. Image Underst.*, 98(1):25–51, 2005.

[8] D. H. Yoo, B. R. Lee, and M. J. Chung. Non-contact eye gaze tracking system by mapping of corneal reflections. In *FGR '02: Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recogniti on*, pages 101–106, Washington, DC, USA, 2002. IEEE Computer Society.