International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459 (Online), An ISO 9001:2008 Certified Journal, Volume 3, Special Issue 1, January 2013)
International Conference on Information Systems and Computing (ICISC-2013), INDIA.

VISION BASED TRACKING SYSTEM

R. Raaghavee[1], R. Sowmya[2], S. Pavithra, Mrs. Dilectin Hepzibah(Guide)[3]
Department of Information Technology, Easwari Engineering College, Chennai, India

Abstract

People's needs are increasing day by day and they want to lead a more sophisticated life where a single movement will help them achieve their tasks. This lead to the development of Artificial Intelligence.AI, a fast emerging technology, uses sensors to communicate directly with the objects and react accordingly. Some examples of AI are Humanoid robots, Google Goggles, driverless cars, eye gaze sensors, head movement trackers and even software that suggests the music a person might like to hear.

This paper emphasizes the study of interaction techniques between user and computer which incorporates eye movement in a convenient way. The widely used eye gaze trackers based on Pupil Center Corneal Reflection (PCCR) technique has two major disadvantages. First, the calibration must be done repeatedly for each and every individual. Second, the tolerance for head movements is very low and it requires the user to hold their head still which is not very comfortable. Therefore, we move for a better technique which overcomes the above disadvantages.

In our proposed system we aim to increase the comfortability of the user by computing the head mapping function that compensates the head movement. In detail, it allows the calibration to be done only once for each person by mapping the eye movement from a reference head position to an arbitrary head position and then the gaze is estimated with respect to the reference head position. With slight modification to the existing eye gaze system, it is bound to help the differently abled people achieve their goals with just a mere look at the screen.

Keywords **— PCCR, eye gaze mapping function.**

I. INTRODUCTION

Eye Gaze is a technique which tracks the eye movement of the user and it replaces keyboard and mouse which were the major inputs to the computer. The early eye tracking system was originally invented by Edmund Huey which was in the form of contact lens with a hole for the pupil. Various research works proves that the eye movement differs for each and every word. The interesting fact is that the users are frequently held by the elements of pictures and also by the texts which has less information. A variety of applications which uses this technique in day-today life includes vehicle simulators, employee training, communication system for the differently abled, product development etc. But the disadvantage of these kind of gazers is that it requires physical contact with the user such as placing a reflective white dot onto the eye [1]or attaching n number of electrodes around the eye[2] and it also expects the user to be motionless.

With a variety of innovations in video cameras, eye tracking technique had been made even better by tracking the video of the eye movements. Since it does not require any physical contact with the user it helps to build an effective non-intrusive eye gaze tracker. Depending upon the various images captured by the camera, various techniques [3], [4], [5], [6], [7], [8], [9] have been proposed to do the eye gaze estimation.

Still, they suffer from two common problems-first, the need for calibration for each user and second, a large restriction on the head motion [10].

Among the various techniques that had been adopted, PCCR is the most widely used technique. The angle of the visual axis angle is calculated by tracking the relative position of the pupil center and a speck of light reflected from the cornea, known as "glint" as shown in Figure 1. The accuracy of the system can be further enhanced by the illumination of the eyes with low level IR which increases the accuracy of the system by producing "bright pupil" effect as shown Figure 1 which in turn makes the video image easy to process.

Several systems [1],[12],[3],[4],[8],[10]which uses this PCCR technique again suffers from the disadvantage of restricting the head movement of the user. In specific, the average error can be less than 1° visual angle, which corresponds to less than 10 mm in the computer screen when the subject is sitting around 550 mm from the computer screen. But if the head moves from the original position from where the calibration has been done then the accuracy of the tracker gradually decreases. Some detailed data are reported about how the calibration mapping function decays as the head moves away from its original position. The only solution to this problem is to perform new calibration. This naturally decreases the efficiency of the system since it is practically impossible to calibrate each and every time when the head moves.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459 (Online), An ISO 9001:2008 Certified Journal, Volume 3, Special Issue 1, January 2013)
International Conference on Information Systems and Computing (ICISC-2013), INDIA.

Fig. 1. Eye image with corneal reflection (or glint).

From the above discussion, it is very obvious that most of the existing gaze tracking systems based on PCCR technique share two common drawbacks: First, the user must perform certain experiments in calibrating the user-dependent parameters before using the gaze tracking system. Second, the user must keep his head uncomfortably still, no significant head movements allowed.

Hence in this paper, a solution is given to above two problems. First, we compute a gaze mapping function before the calibration process, so that the user need not calibrate each and every time as the head moves. Second, the user can move his head freely which in turn increases the user-system interface. Therefore, by using our gaze tracking technique, a more robust, accurate, comfortable and useful system can be built.

II. PCCR TECHNIQUE

The PCCR technique has two major components:

1) Pupil-glint vector extraction
2) Gaze mapping function acquisition.

2.1 Pupil-Glint Vector Extraction:

The estimation of the gaze starts along with the pupil-glint vector extraction .The pupil center and the glint center are extracted accurately from the eye image that is caught on the camera using computer visualization techniques[7],[15]. The pupil center and the glint center are then connected to form a 2D pupil glint vector *v* as shown in Figure3.

2.2. Specific Gaze Mapping Function Acquisition:

After obtaining the pupil-glint vectors, a calibration process is done to map the extracted pupil-glint vector to the user's fixation point in the screen for current head position by using a specific gaze mapping function.

The obtained pupil-glint vector *v* is represented as (*vx*, *vy*) while the screen gaze point *Ss* is represented by (*xgaze*, *ygaze*) in the screen coordinate system.

The specific gaze mapping function $Ss = f(v)$ can be modeled by the following nonlinear equations :

$$\left.\begin{aligned} x_{gaze} &= a_0 + a_1 * v_x + a_2 * v_y + a_3 * v_x * v_y \\ y_{gaze} &= b_0 + b_1 * v_x + b_2 * v_y + b_3 * {v_y}^2 \end{aligned}\right\} \quad (1)$$

$a0$, $a1$, $a2$, $a3$ and $b0$, $b1$, $b2$, $b3$ represent the coefficients that are estimated from a set of pairs of pupil-glint vectors and the corresponding screen gaze points and these pairs are collected from the calibration process. During the calibration, the user is instructed to visually follow a shining dot that is displayed at various predefined locations on the computer screen. In addition to this, the user must also keep his head as still as possible.

The specific gaze mapping function can be used to estimate the user's gaze point in the screen accurately based on the extracted pupil-glint vector, if the user does not move his head significantly after gaze calibration. Incase if the user moves his head away from the position where the specific gaze calibration is performed, the specific gaze mapping function will fail to estimate the gaze point accurately because of the pupil glint vector changes caused by the head movement.

2.2.1. Head Motion Effects On Pupil-Glint Vector

The following figure 2 shows the ray diagram of the pupil-glint vector generation when an eye is located at two different 3D positions $O1$ and $O2$ in front of the camera due to head movement. In a simpler way, the eye is represented by a cornea, the cornea is modeled as a convex mirror and the IR light source used to generate the glint is located at O, which are applicable to all the subsequent figures. Assume that the origin of the camera is located at O, $p1$ and $p2$ are the pupil centers and $g1$ and $g2$ are the glint centers generated in the image. Further, at both positions, the user is looking at the same point of the computer screen S. According to the light ray diagram shown in Figure 2, the generated pupil-glint vectors $g{-}{-}1 \rightarrow p1$ and $g{-}{-}2 \rightarrow p2$ will be significantly different in the images as shown in Figure 3. There are two factors which are responsible for this pupil glint vector difference: first, the eyes are at different positions in front of the camera; second, eyes at different positions rotate themselves differently in order to look at the same screen point.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459 (Online), An ISO 9001:2008 Certified Journal, Volume 3, Special Issue 1, January 2013)
International Conference on Information Systems and Computing (ICISC-2013), INDIA.

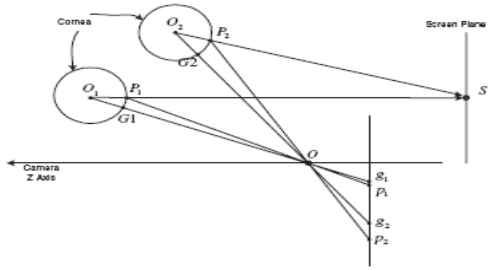

Fig. 2. Pupil and glint image formations when eyes are located at different positions while gazing at the same screen point (side view).

(a) eye image at location O_1 (b) eye image at location O_2

Fig. 3. The pupil-glint vectors generated in the eye images when the eye is located at O_1 and O_2 in Figure 2.

As the head moves, the eye also moves. Therefore, a set of two different pupil-glint vectors in the image will be generated since the user is looking at a fixed point on the screen while moving his head in front of the camera . In case, if it is not corrected, inaccurate gaze points will be produced after inputting them into the specific gaze mapping function obtained at the reference eye position. Therefore, the head movement effects on these pupil-glint vectors must be eliminated in order to utilize the specific gaze mapping function to estimate the screen gaze points more accurately. Hence, a better technique is proposed to eliminate the head movement effects on these pupil-glint vectors before inputting them into the specific gaze mapping function. This technique, predicts whether the user has moved his head or not by estimating accurate gaze screen points.

III. Head Compensation Model

3.1. Approach Overview

First, we find a gaze mapping function f_{o1} at a reference position o1, between the pupil glint vector v1 and the screen coordinate s via gaze calibration procedure using equations 1. The function $fO1$ can be expressed as follows:

$$S = f_{O1}(v1) \qquad (2)$$

Let us assume that a pupil-glint vector v2 will be produced in the image when the eye moves to a new position O2 as the head moves while the user is looking at the same screen point *S*.

If O2 is entirely different from O1, then v2 cannot be used to estimate the screen gaze point due to the changes of the pupil glint vector caused by the head movement in f_{o1}. Corrected pupil-glint vector v2' can be obtained if the changes caused by the head movement can be eliminated. Further, this corrected pupil-glint vector *v2'* is the generated pupil-glint vector *v*1 of the eye at the position *O1* when gazing at the same screen point *S*. Therefore, the mapping function g between two different pupil-glint vectors at two different head positions when still gazing at the same screen point can be written as follows:

$$v2' = g(v2,O2,O1) \qquad (3)$$

Where *v2'* is the corrected pupil-glint vector with respect to the initial reference head position *O*1.Form this, the screen gaze point can be estimated accurately via the specific gaze mapping function *fO*1 as follows:

$$S = f_{O1}\,(g(v2,O2,O1)) = F(v2,O2) \qquad (4)$$

Where the function *F*, generalized gaze mapping function, provides the gaze mapping function for a new eye position *O*2 dynamically. Using our proposed technique, whenever the head moves, a gaze mapping function at each new 3D eye position will be automatically estimated which solves the issue of the head movement through *g*.

3.2. Head Movement Compensation

Firstly, we find the gaze mapping function g in this process. The formation of pupil-glint vector in the image for an eye in front of the camera is shown in Figure 4. Two different pupil-glint vectors *g*1*p*1 and *g*2*p*2 are generated in the image if the eye is located at two different positions *O*1 and *O*2 while gazing at the same screen point *S*. Further, as shown in Figure 4, a parallel plane *A* of the image plane that goes through the point *P*1 will intersect the line *O*1*O* at *G*1. Another parallel plane *B* of the image plane that goes through the point *P*2 will intersect the line *O*2*O* at *G*2. From this we can tell that, *g*1*p*1 is the projection of the vector *G*1*P*1 and *g*2*p*2 is the projection of the vector *G*2*P*2 in the image plane. The vectors *g*1*p*1, *g*2*p*2, *G*1*P*1 and *G*2*P*2 can be represented as 2D vectors in the $X - Y$ plane of the camera coordinate system since plane *A*, plane *B* and the image plane are parallel.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459 (Online), An ISO 9001:2008 Certified Journal, Volume 3, Special Issue 1, January 2013)
International Conference on Information Systems and Computing (ICISC-2013), INDIA.

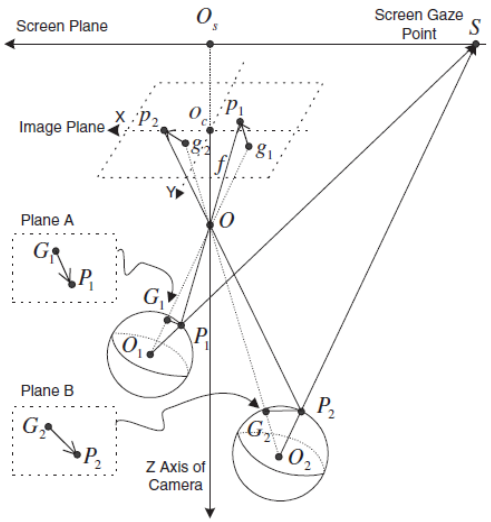

Fig. 4. Pupil and glint image formation when the eye is located at different positions in front of the camera.

3.2.1. Image Projection Of Pupil-Glint Vector

Let us assume that the 3D pupil centers $P1$ and $P2$ are represented as ($x1$, $y1$, $z1$) and ($x2$, $y2$, $z2$) and the glint centers $g1$ and $g2$ are represented as ($xg1$, $yg1$, $-f$) and ($xg2$, $yg2$, $-f$), where focal length of the camera f, and the screen gaze point S is represented by (xs, ys, zs). The image projection of the pupil-glint vectors via the pinhole camera model, can be expressed as follows:

$$\overrightarrow{g_1p_1} = -\frac{f}{z_1} * \overrightarrow{G_1P_1} \quad (5)$$

$$\overrightarrow{g_2p_2} = -\frac{f}{z_2} * \overrightarrow{G_2P_2} \quad (6)$$

Let us consider that the vectors $G1P1$ and $G2P2$ are represented as ($Vx1$, $Vy1$) and ($Vx2$, $Vy2$) respectively and the projected vectors $g1p1$ and $g2p2$ are represented as ($vx1$, $vy1$) and ($vx2$, $vy2$) respectively. By combining the equations 5 and 6, the following equation can be derived:

$$v_{x1} = \frac{V_{x1}}{V_{x2}} * \frac{z_2}{z_1} * v_{x2} \quad (7)$$

$$v_{y1} = \frac{V_{y1}}{V_{y2}} * \frac{z_2}{z_1} * v_{y2} \quad (8)$$

As the head moves in front of the camera, the changes in pupil-glint vector can be described by using the above two equations. It is obvious from the above two equations that each component of the pupil-glint vector can be transformed individually.

Therefore, the derivation of equation 7 for X component will be done in the next section.

3.2.2. First Case: The Cornea Centre And The Pupil Centre Lie On The Camera's X − Z Plane:

When the corneal centre and the pupil centre of an eye lies on the X-Z plane of the camera coordinate system, the formation of pupil-glint vector is shown by the ray diagram in the Figure 5. Thus, we can represent either the generated pupil-glint vectors $p1g1$ and $p2g2$ or the vectors $P1G1$ and $P2G2$ as one dimensional vectors, $p1g1 = vx1$, $p2g2 = vx2$, $P1G1 = Vx1$ and $P2G2 = Vx2$.

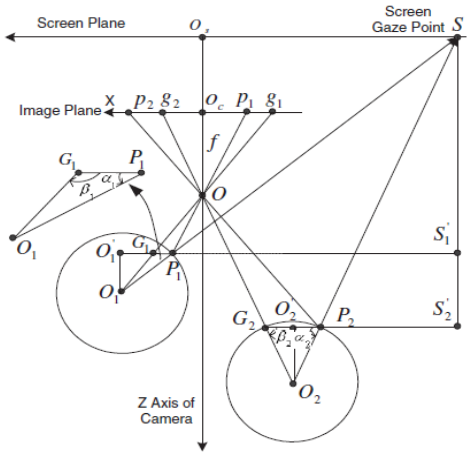

Fig. 5. Pupil and glint image formation when the eye is located at different positions in front of the camera (Top-down view).

From the Figure 5, we can represent the vectors G_1P_1 and G_2P_2 as follows:

$$\overrightarrow{G_1P_1} = \overrightarrow{G_1O_1'} + \overrightarrow{O_1'P_1} \quad (9)$$

$$\overrightarrow{G_2P_2} = \overrightarrow{G_2O_2'} + \overrightarrow{O_2'P_2} \quad (10)$$

In order to provide simplicity we are using $r1$ and r2 to represent the length of O_1P_1 and O_2P_2, and α1, α2 and $\beta1$, $\beta2$ are used to represent the angles $\angle G_1P_1O_1$, $\angle G_2P_2O_2$ and $\angle P_1G_1O_1$, $\angle P_2G_2O_2$ respectively. Therefore by using the geometries shown in Figure 5, the vectors G_1P_1 and G_2P_2 can be further achieved as follows:

$$\overrightarrow{G_1P_1} = -\frac{r_1 * sin(\alpha 1)}{tan(\beta 1)} - r_1 * cos(\alpha 1) \quad (11)$$

$$\overrightarrow{G_2P_2} = -\frac{r_2 * sin(\alpha 2)}{tan(\beta 2)} - r_2 * cos(\alpha 2) \quad (12)$$

According to Figure 5, line G_1P_1 and line G_2P_2 are parallel to the X axis of the camera. Therefore, $tan(\beta 1)$ and $tan(\beta 2)$ can be obtained from the rectangles g_1O_cO and g_2O_cO individually as follows:

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459 (Online), An ISO 9001:2008 Certified Journal, Volume 3, Special Issue 1, January 2013)
International Conference on Information Systems and Computing (ICISC-2013), INDIA.

$$tan(\beta 1) = \frac{f}{\overrightarrow{o_c g_1}} \quad (13)$$

$$tan(\beta 2) = \frac{f}{\overrightarrow{o_c g_2}} \quad (14)$$

Where, g_1 and g_2 are the glints in the image, and O_c is the principal point of the camera. To make the process simple, xg_1 and xg_2 are chosen to represent $O_cg_1$and O_cg_2 respectively. Thus $tan(\beta 1)$ and $tan(\beta 2)$ can be obtained accurately, after detecting the glints in the image. Moreover, $sin(\alpha 1)$, $cos(\alpha 1)$, $sin(\alpha 2)$ and $cos(\alpha 2)$ can be obtained from the geometries of the rectangles $P_1SS'_1$ and $P_2SS'_2$ directly. Therefore, equations 11 and 12 can be derived as follows:

$$V_{x1} = r_1 * \frac{(z_s - z_1) * x_{g1}}{P_1S * f} + r_1 * \frac{(x_s - x_1)}{P_1S} \quad (15)$$

$$V_{x2} = r_2 * \frac{(z_s - z_2) * x_{g2}}{P_2S * f} + r_2 * \frac{(x_s - x_2)}{P_2S} \quad (16)$$

3.2.3. Second Case: The Cornea Center And The Pupil Center Do Not Lie On The Camera's $X-Z$ Plane:

In real time, the cornea centre and the pupil centre do not always lie on the same X - Z plane. Therefore, by projecting the ray diagram given in the Figure 4 into X – Z plane along the Y axis of the camera's coordinate system, we can obtain the ray diagram as shown in Figure 5. The point P_1, O_1, S is the projection of the pupil centre, cornea centre, and the screen gaze point S' in the $X - Z$ plane as shown in the Figure 6. It can be seen from the Figure 6 that from the starting point O_{c1}, a parallel line $O_{c1}P_1$' of line O_1P_1 intersects with line Pc_1P_1 at P_1' and also starting from Pc_1, a parallel line Pc_1O_1'' of line P_1S intersects with line SS' at $O1$''. Because $O_{c1}Pc_1$ represents the distance r between the pupil centre to the cornea center, which will not change as the eyeball rotates, O_1P_1 can be derived as follows:

$$r_1 = O_1P_1 = r * \frac{P_1S}{\sqrt{P_1S^2 + (y_1 - y_s)^2}} \quad (17)$$

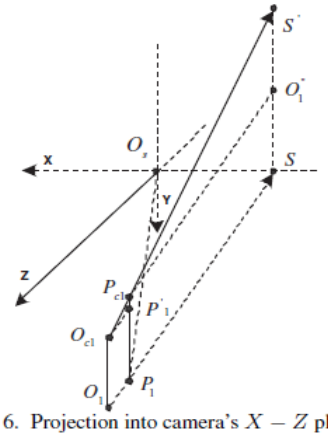

Fig. 6. Projection into camera's $X - Z$ plane.

Thus, if the eye moves to a new position $O2$ as shown in Figure 5, $O2P2$ can be represented as follows:

$$r_2 = O_2P_2 = r * \frac{P_2S}{\sqrt{P_2S^2 + (y_2 - y_s)^2}} \quad (18)$$

Substitute the formulae for r1 and r2 into equations 15 and 16, we obtain V_{x1}/V_{x2} as follows:

$$\frac{V_{x1}}{V_{x2}} = d * \frac{[(z_s - z_1) * x_{g_1} + (x_s - x_1) * f]}{[(z_s - z_2) * x_{g_2} + (x_s - x_2) * f]} \quad (19)$$

where d is set as follows:

$$d = \frac{\sqrt{(z_2 - z_s)^2 + (x_2 - x_s)^2 + (y_2 - y_s)^2}}{\sqrt{(z_1 - z_s)^2 + (x_1 - x_s)^2 + (y_1 - y_s)^2}}$$

Thus from the above equations, we can obtain the equations 7 and 8 as follows:

$$v_{x1} = d * \frac{[(z_s - z_1) * x_{g_1} + (x_s - x_1) * f]}{[(z_s - z_2) * x_{g_2} + (x_s - x_2) * f]} * \frac{z_2}{z_1} * v_{x2} \quad (20)$$

$$v_{y1} = d * \frac{[(z_s - z_1) * y_{g_1} + (y_s - y_1) * f]}{[(z_s - z_2) * y_{g_2} + (y_s - y_2) * f]} * \frac{z_2}{z_1} * v_{y2} \quad (21)$$

The above equations constitute the head mapping function g between two different pupil-glint vectors at two different head positions when still gazing at the same screen point.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459 (Online), An ISO 9001:2008 Certified Journal, Volume 3, Special Issue 1, January 2013)
International Conference on Information Systems and Computing (ICISC-2013), INDIA.

3.3. Iterative Algorithm for Gaze Estimation:

However, unless the gaze point $S = (xs, ys, zs)$ of the screen is known, the derived equations 20 and 21 cannot be established for the head mapping function. As a result, the gaze point S is also a variable of the head mapping function g, which can be further expressed as follows:

$$v_2' = g(v_2, P_2, P_1, S) \quad (22)$$

As per the description of calibration procedure in the section 2 let us consider that a specific gaze mapping function f_{P1} is known. After integrating the head mapping function g into the specific gaze mapping function f_{P1} via equation 4, the generalized gaze mapping function F can be rewritten as follows:

The above equation 23 becomes a recursive function given the extracted pupil-glint vector $v2$ from the eye

$$S = F(v_2, P_2, S) \quad (23)$$

image and the new location of the eye $P2$. Therefore, we provide an iterative function to solve this problem. The screen centre $S0$ is chosen as an initial gaze point and then a corrected pupil-glint vector $v2'$ can be obtained from the detected pupil-glint vector $v2$ via the head mapping function g. By giving the corrected pupil-glint vector v2' as the input to the specific gaze mapping function f_{P1}, a new screen gaze point S' can be estimated. A new corrected pupil-glint vector $v2'$ can be obtained by using S'.

The loop stops if and only if the estimated screen gaze point does not change anymore. However, the whole iteration process will converge in less than 5 loops, which is very fast. Thus the gaze point can be estimated accurately by using our proposed method.

IV. Experiment Results

4.1. System Setup

With this new algorithm we tried to see what result was produced and its effectiveness . The system has two cameras mounted beneath the monitor screen. An Infra Red light illuminator is mounted at the centre of the lens in any one of the camera. This will produce the corneal glint in the image. To estimate the gaze, pupil glint vectors extracted from the images captured by the camera is used. With the modification to the pre existing algorithm we used 3D positions to get accurate gaze position. To get the 3D position, another camera is mounted close to the camera which has IR light illuminator. From the output of these two cameras, which gives a stereo vision system we can obtain the 3D position accurately.

But our Eye Tracker system needs to know the 3D representation of monitor screen plane in the coordinate system of the camera with the IR illuminator. This representation is done via the calibration method introduced in [14]. Once the calibration is done, the configuration between camera and monitor will not change. It is fixed during eye tracking. This is the main advantage of our eye tracking system. It requires calibration to be done only once and this remains constant.

4.2. System validation

In this section we try to validate our system and check if the results produced match the expected results. The equations 20 and 21 of the head mapping function is validated by performing the following experiment. We chose a screen point Sc = (132.75, −226.00,−135.00) as the gaze point. The user was made to look at this point from twenty different positions by standing in front of the camera; as the user looks at each location, the pupil-glint vector and the 3D pupil centre data are collected. The 3D pupil centers and the pupil-glint vectors of the first two samples P1,P2 are shown in Table I, whereP1 is taken as the reference position. The second column indicates the original pupil-glint vectors, while the third column indicates the transformed pupil-glint vectors by the head mapping function which was caused due to the change in position of user's head. Transformation error is the difference between the transformed pupil-glint vector of P2 and the reference pupil-glint vector at P1. Figure 7 illustrates the transformation errors for all the twenty samples which are the twenty different locations from which user gazed the camera . From the table ,It is observed that the average transformation error is just around 1pixel which is not a big error. From this observation, the validity of our proposed system is found. Our system which has error of one pixel has more advantage than the other systems

TABLE II
GAZE ESTIMATION ACCURACY

Distance to the camera (mm)	Horizontal Accuracy (degrees)	Vertical Accuracy (degrees)
340.26	0.68	0.83
400.05	1.31	1.41
462.23	1.54	1.90
552.51	1.73	2.34

4.3. Accuracy of Eye Gaze System

After validating the system , we determined the accuracy of this system. The drawback of the system is that when user moves away from the camera, the eye in the image of the camera becomes smaller.

This increases the pixel measurement error caused by lower image resolution. Due to this, the accuracy of our eye gaze tracker will decrease if user walks away from camera. In this experiment we analyze the effect of the distance to the camera on the accuracy of our system. A test user was made to perform the calibration when he was sitting at 330 mm distance from the camera. Once the calibration is done, the user was made to stand at four different positions, which was located at different distances from the camera. The different distances are mentioned in Table II. At each of the location, the user followed the moving objects that displays 12 predefined positions on the screen. Table II shows the accuracy of gaze estimation at four different locations the user was positioned. This table shows that as the user moves away from camera, the resolution of gaze is decreasing. But within the space allocated for head movement, which is approximately 200x200x300 mm which is at a distance of 450mm from the camera , the average horizontal angular accuracy is around 1.3 and average vertical angular accuracy is 1.7. This accuracy is acceptable for the computer interactive applications. This accuracy value is very high when compared with other systems where users have to strain their eyes to make an action occur. In addition to this, the space allocated for the head movement is very large. This allows the user to sit comfortably in front of camera and interact with the computer naturally.

V. CONCLUSION

In this paper, a solution has been proposed to overcome the drawbacks of classical PCCR eye gaze tracking technique. Though PCCR technique has many advantages like high accuracy and stability , it restricts head movement. This makes the user to sit in a stiff position and makes them uncomfortable. But with this system, the user is free to move his head inside the allotted space . Another disadvantage in the existing system is the calibration which needs to be done repeatedly for the same user. With the new algorithm, the calibration is done just once for every new user. This technique is possible only because of the head mapping function which automatically accommodates the head movement changes. Thus a better eye gaze system is introduced to enhance the features of AI.

REFERENCES

[1] S. Milekic, "The more you look the more you get: intention-based interface using gaze-tracking," in Bearman, D., Trant, J.(des.) Museums and the Web 2002: Selected papers from an international onference, Archives and Mseum Informatics, 2002.

[2] K. Hyoki, M. Shigeta, N. Tsuno, Y. Kawamuro, and T. Kinoshita, "Quantitative electro-oculography and electroencephalography as indices of alertness," in Electroencephalography and Clinical Neurophysiology, 1998, pp. 213–219.

[3] Y.Ebisawa, M. Ohtani, and A. Sugioka, "Proposal of a zoom and focus control method using an ultrasonic distance-meter for video-based eye-gaze detection under free-hand condition," in Proceedings of the 18th Annual International conference of the IEEE Eng. in Medicine and Biology Society, 1996.

[4] C.H. Morimoto, D. Koons, A. Amir, and M. Flickner, "Frame-rate pupil detector and gaze tracker," in IEEE ICCV'99 Frame-rate Workshop, 1999.

[5] A. T. Duchowski, "Eye tracking methodology: Theory and practice," in Spring Verlag, 2002.

[6] R.J.K. Jacob and K. S. Karn, "Eye tracking in human-computer interaction and usability research: Ready to deliver the promises," 2003, Oxford, Elsevier Science.

[7] D. H. Yoo, "Non-contact eye gaze estimationn system using robsut feature," CVIU, special Issue on eye detection and tracking, 2005.

[8] LC Technologies Inc, "The eye gaze development system, http://www.eyegaze.com," .

[9] Z. Zhu and Q. Ji, "Eye and gaze tracking for interactive graphic display," Machine Vision and Applications, vol. 15, no. 3, pp. 139–148, 2004.

[10] C.H. Morimoto and M. R.M. Mimica, "Eye gaze tracking techniques for interactive applications," CVIU, special issue on eye detection and tracking, 2005.

[11] T.E. Hutchinson, K.P. White Jr., K.C. Reichert, and L.A. Frey, "Human-computer interaction using eye-gaze input," in IEEE Transactions on Systems, Man, and Cybernetics, 1989, pp. 1527–1533.

[12] R.J.K. Jacob, "Eye-movement-based human-computer interaction techniques: Towards non-command interfaces," 1993, pp. 151–190, Ablex Publishing corporation, Norwood, NJ.

[13] Z. Zhu and Q. Ji, "Robust real-time eye detection and tracking under variable lighting conditions and various face orientations," CVIU, special issue on eye detection and tracking, 2005.

[14] S. W. Shih and J. Liu, "A novel approach to 3-d gaze tracking using stereo cameras," in IEEE Trans. Syst. Man and Cybern., part B, 2004, number 1, pp. 234–245.