

A single-camera gaze tracking system under natural light

Feng Xiao
Institute of VLSI Design,
Zhejiang University, China

Dandan Zheng
Institute of VLSI Design,
Zhejiang University, China

Kejie Huang
Institute of VLSI Design,
Zhejiang University, China

Yue Qiu
Institute of VLSI Design,
Zhejiang University, China

Haibin Shen
Institute of VLSI Design,
Zhejiang University, China

Gaze tracking is a human-computer interaction technology, and it has been widely studied in the academic and industrial fields. However, constrained by the performance of the specific sensors and algorithms, it has not been popularized for everyone. This paper proposes a single-camera gaze tracking system under natural light to enable its versatility. The iris center and anchor point are the most crucial factors for the accuracy of the system. The accurate iris center is detected by the simple active contour snake, which is initialized by the prior knowledge of eye anatomical dimensions. After that, a novel anchor point is computed by the stable facial landmarks. Next, second-order mapping functions use the eye vectors and the head pose to estimate the points of regard. Finally, the gaze errors are improved by implementing a weight coefficient on the points of regard of the left and right eyes. The feature position of the iris center achieves an accuracy of 98.87% on the G4E database when the normalized error is lower than 0.05. The accuracy of the gaze tracking method is superior to the state-of-the-art appearance-based and feature-based methods on the EYEDIAP database.


Keywords: Eye movement, eye tracking, gaze, usability, single-camera, facial landmark, iris center, anchor point, head pose, mapping functions

Introduction

Gaze tracking is a kind of human-computer interaction technology that creates an easy and effective interaction for serving the disabled, learning, entertainment, etc. Meanwhile, it is also a research tool, and it has been widely used in marketing studies (Lahey & Oxley, 2016), reading research (Huck, Thompson, Cruice, & Marshall, 2017), and so forth. Gaze tracking techniques can be

divided into electrooculography-based, coils-based, and video-based (infrared and natural light) techniques and so on (Holmqvist & Andersson, 2017). The third technique is less intrusive than the first two, which require physical contact sensors such as electrodes and scleral coils.

Today, a variety of existing remote video-based gaze tracking systems under infrared (IR) light in academia and industry have achieved accurate results. For instance, the Dual-Purkinje-Image (DPI) gaze tracker (Crane & Steele, 1985) achieves an accuracy better than 0.1° (Deubel & Schneider, 1996). The Eyelink 1000 system performs at an accuracy below 0.5° with a white background (Drewes, Zhu, Hu, & Hu, 2014). However, infrared sources are sensitive to ambient light. IR gaze trackers also have reflection problems when people wear glasses.

Received November 18, 2017; Published September 20, 2018.
Citation: Xiao, F., Zheng, D., Huang, K., Qiu, Y., & Shen, H. (2018). A single-camera gaze tracking system under natural light. *Journal of Eye Movement Research*, 11(4):5.
Digital Object Identifier: 10.16910/jemr.11.4.5
ISSN: 1995-8692
This article is licensed under a [Creative Commons Attribution 4.0 International license](https://creativecommons.org/licenses/by/4.0/). 

Therefore, development of a gaze tracking system under natural light has become an increasingly important field of research.

Recently, video-based gaze tracking system under natural light are capable of tracking the gaze. However, some of them rely on multiple cameras (Pan & Steed, 2014), High-Definition (HD) cameras (El Hafí, Ding, Takamatsu, & Ogasawara, 2017) and RGB-D cameras (Ghiass & Arandjelovic, 2016; J.Li & S.Li, 2016), which limit their applications. With the popularity of cameras, gaze tracking with a single-camera under natural light becomes a research hot spot, but one of the major challenges is the requirement for an accurate gaze tracking algorithm. Therefore, we concentrate on a regression-based gaze tracking system with a single-camera under natural light in this paper.

The accuracy of regression-based gaze tracking is directly influenced by the eye vectors that are derived from the iris centers and the facial stable point (anchor point or reference point). However, the performance of various iris/pupil center localization methods significantly degrades in low resolution images because of interference such as glass/iris reflection, and eyelid. In addition, the anchor points of the eye corners vibrate with eye rotation (Sesma, Villanueva, & Cabeza, 2012) and can be blocked due to large head movements. Therefore, an accurate localization method in low resolution images (Xiao, Huang, Qiu, & Shen, 2018) is improved to detect the iris center. Then, a novel anchor point is proposed to overcome the drawbacks mentioned above. Finally, the Points of Regard (POR) of the left and right eyes are combined to improve the accuracy of the system. Compared with other regression-based methods, the main contributions in this paper are listed in the following:

- (1) An accurate feature position localization method for the iris center is implemented in low resolution images by combining facial landmarks, the prior knowledge of eye anatomical dimensions, and the simple active contour snake (Thevenaz & Unser, 2008).

- (2) A novel anchor point is computed by averaging the stable facial landmarks, which improves the accuracy of the gaze tracking system.

- (3) A weight coefficient is used on the POR of the left and right eyes to revise the final POR, which reduces the error of the gaze tracking.

The rest of the paper is structured as follows: In the next section, the related work is presented. The details of the proposed method are covered in the *Methods* section. The evaluation of the proposed scheme and statistical results on public databases are shown in the *Evaluation* section. The discussion is presented in the final section.

Related work

This section overviews gaze tracking systems under natural light. The systems can be classified into feature-based and appearance-based methods (Hansen & Ji, 2010).

Feature-based methods

Feature-based methods extract features such as the iris/pupil center, eye corners and iris/pupil contours. Then, model-based and regression-based methods use the features to track the gaze. Model-based methods (Wood & Bulling, 2014; J.Li & S.Li, 2016) use a geometric eye model to compute the gaze direction from the features. Regression-based methods (Valenti, Staiano, Sebe, & Gevers, 2009; Skodras, Kanas, & Fakotakis, 2015) compute a mapping function between the gaze direction and eye vectors.

The performance of model-based methods relies on the accurate detection of the iris center. In J.Li and S.Li (2016), the iris center was obtained by an ellipse fitting algorithm, where the ellipse of the iris in the image was described by the yaw and pitch angles. J.Li and S.Li (2016) achieved 7.6° and 6.7° in horizontal and vertical directions on the public EYEDIAP database (Mora, Monay, & Odobez, 2014) with an execution speed of 3 frames per second (fps) on a 2.5-GHz Inter(R) Core(TM) i5-2400S processor. In Wood and Bulling (2014), the shape of the iris was estimated by ellipse fitting. Then, an accuracy of 7° of the gaze direction was inferred by the hypothesis that the shape of the iris appears to deform from circular to elliptical when the iris orientation changes. Wood and Bulling (2014) achieved an execution speed of 12 fps on a commodity tablet computer with a quad-core 2 GHz processor. Ellipse fitting has a low consistency and reliability because iris edges or points cannot be accurately extracted in low resolution images.

In addition to the iris center, the anchor point is one of the key features influencing the accuracy of the regression-based methods. In Valenti et al. (2009), the eye

corner was used as the anchor point. Instead of detecting the eye corners, the anchor point in Skodras et al. (2015) was set as the center coordinate of the patch which contains the inner eye corners and eyebrow edges. The proposed system yielded a mean accuracy of 2.33° and 1.8° in the horizontal and vertical directions on their self-built database and 7.53° on the public UulmHPG database (Weidenbacher, Layher, Strauss, & Neumann, 2007). However, eye corners or the center of the patch cannot be accurately detected in low resolution images with a large head rotation.

Appearance-based methods

Appearance-based methods do not extract specific features and usually learn a mapping function from eye images to gaze directions. In Sugano, Matsushita, & Sato (2014), gaze estimation was learned by random regression forests with a significantly larger dataset, which reduced the error by 50% from the work in Mora and Odobez (2012) with an error larger than 10° . In Ghiass and Arandjelovic (2016), k-nearest neighbor regression and adaptive linear regression were used to learn mapping functions between eye images and gaze directions, which achieved a mean accuracy of 7.2° (keeping the head still) and 8.9° (head movement) on the EYEDIAP database. With the development of deep learning, convolutional neural networks (CNNs) have been used to estimate the gaze with millions of eye images in Krafka, et al. (2016). They proved that a largescale dataset and a large variety of data could improve the accuracy of the appearance-based model for gaze tracking, which achieved errors of 1.71 cm and 2.53 cm without calibration on mobile phones and tablets, respectively. Krafka, et al. (2016) achieved a detection rate of 10–15 fps on a typical mobile device. One of the main drawbacks to appearance-based methods is that the appearance of the eyes is significantly affected by the head pose (Skodras et al., 2015). In addition, compared with feature-based methods, appearance-based methods generally require larger numbers of training images.

Methods

The flow chart of the gaze tracking system is depicted in Figure 1. The system includes calibration and testing phases. In the calibration phase, mapping functions are regressed by the head pose, eye vectors and gaze direc-

tions. Afterwards, the head pose, eye vectors and regressive mapping functions are used to track the gaze in the testing phase. The feature extraction consists of three parts for the iris centers, anchor point and head pose calculations.

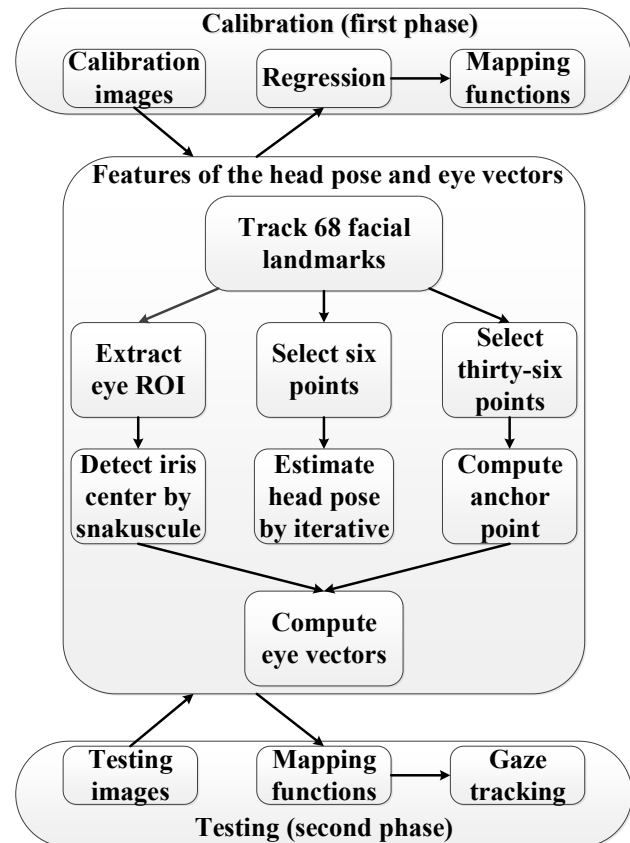


Figure 1. Flow chart of the gaze tracking system

First, the eye Region of Interests (ROIs) are extracted by twelve points around the eyes, which are tracked by the facial landmarks algorithm in Kazemi and Sullivan (2014). The eye ROIs are resized to twice the original sizes. Then, greyscale erosion is applied. After that, the snakuscule is used to locate the iris centers.

Second, thirty-six stable facial landmarks are used to compute the anchor point. Thereafter, the left and right eye vectors are computed by the iris centers and the anchor point, respectively.

Third, the head pose is estimated based on the six facial landmarks of eye corners, nose tip, mouth corners and chin by using the OpenCV (Kaehler & Bradski, 2016) iterative algorithm.

Details of the overall system are discussed in the following subsections.

Iris center localization

The eye ROI should be detected before locating the iris center. Facial landmarks can provide more precise positioning of the mouth, eyes, nose, etc. Therefore, an ensemble of regression trees algorithm (Kazemi & Sullivan, 2014) is used to detect 68 facial landmarks. This algorithm uses intensity differences between pixels to estimate the positions of 68 facial landmarks. The locations of 68 facial landmarks are shown in Figure 2.

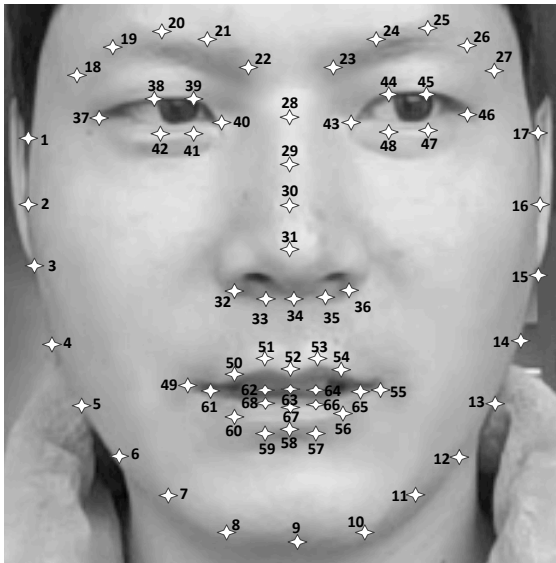


Figure 2. The locations of 68 facial landmarks

The rectangular eye ROIs are next extracted by the twelve points around the eyes. The boundary coordinates of the eye ROIs are computed by the equations in Table 1.

Table 1. The boundary coordinates of eye ROIs.

Left eye	Right eye
$X_l = P_{43x}$	$X_r = P_{37x}$
$X_r = P_{46x}$	$X_r = P_{40x}$
$Y_t = \min\{P_{44y}, P_{45y}\} - 3$	$Y_t = \min\{P_{38y}, P_{39y}\} - 3$
$Y_b = \max\{P_{47y}, P_{48y}\} + 3$	$Y_b = \max\{P_{41y}, P_{42y}\} + 3$

Note: X_l , X_r , Y_t and Y_b are the left, right, top and bottom coordinates of the eye ROIs. P_{ix} and P_{iy} are respectively the x and y coordinates of the i^{th} facial landmark. $\max\{\cdot\}$ and $\min\{\cdot\}$ denote taking the maximum and minimum values respectively among the two values. The coordinate origin is in the top left corner of the image.

Results in Figure 3 show that accurate eye ROIs can be extracted even when large head rotations occur. Then, the eye ROIs dimensions are magnified by a factor of two. Grayscale erosion with a 1-pixel disk structure element is used in the eye ROIs to delete possible noise. Finally, the simple active contour snake is used to locate the iris centers.



Figure 3. Eye ROIs extraction on the EYEDIAP database

As shown in Figure 4, snakuscule is an area-based circular snake that contains an outer annulus and an inner disk. It performs well in detecting circular regions with the maximum gray difference of the outer annulus and the inner disk. β (Figure 4) is defined as the ratio of outer to inner radius.

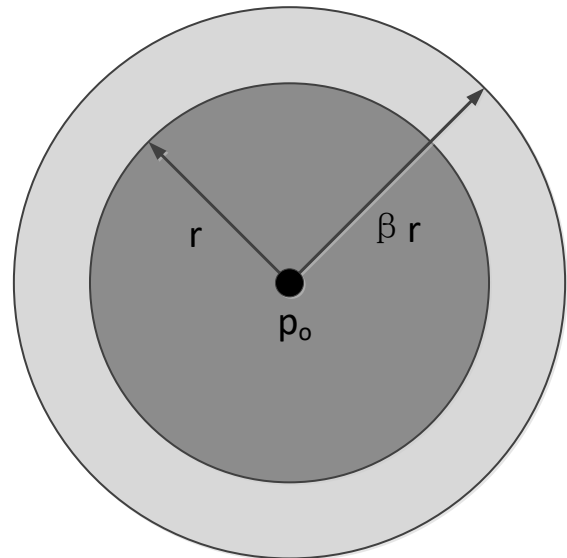


Figure 4. Structure of a snakuscule

Based on the sclera and the iris having the maximum gray difference, snakusculc can expand or shrink to maximize the values of the outer annulus and the inner disk. However, uncontrolled expansion or shrinkage of the snakusculc needs numerous iterations before its final convergence. To overcome the shortcoming, the snakusculc's inner radius is initialized by the eye anatomical dimensions that the radius of the eyeball is in the range of 12-13 mm (Kim & Ramakrishna, 1999) and the radius of the iris is approximately equal to an anatomical constant (approximately 7 mm) (Newman, Matsumoto, Rougeaux, & Zelinsky, 2000) for most people. In addition, the method works well as the width of the eye ROI extracted by facial landmarks is close to the diameter of the eyeball in the image. Therefore, the snakusculc inner radius is initialized by

$$r = [N \times \alpha] \quad (1)$$

where N is the width of the eye ROI and α is a constant that involves the ratio of iris radius to the eye ROI width.

Using the initialized snakusculc, the gray difference of the outer annulus and the inner disk is calculated by use of formula (2) is suggested in Xiao et al. (2018).

$$G(p_o) = \frac{\iint_{r < \|p - p_o\| \leq \beta r} f(p) dx dy}{\iint_{r < \|p - p_o\| \leq \beta r} dx dy} - \frac{\iint_{\|p - p_o\| \leq r} f(p) dx dy}{\iint_{\|p - p_o\| \leq r} dx dy} \quad (2)$$

where $f(p)$ denotes the image gray value at the position p . Formula (2) is used to compute the gray difference of $G(p_i)$, where $p_i = (x_i, y_i)$, $x_i \in [\beta r, N - \beta r]$, x_i is an integer with a minimum interval of 1, $y_i = [M/2]$ and M is the height of the eye ROI. In other words, the snakusculc is used to compute gray differences from left to right along the horizontal centerline in the eye ROI. The location $p_{rc}(x_{rc}, y_{rc})$ with the maximum $G(p_i)$ is the rough iris center.

As shown in Figure 5, $(2\delta+1) \times (2\delta+1)$ iris center candidate points are determined by the rough iris center (x_{rc}, y_{rc}) in the eye ROI. The unit of δ is the pixel. The iris center candidate points are used to accurately locate the iris center. In other words, in the range of $[x_{rc} \pm \delta, y_{rc} \pm \delta]$, $(2\delta+1) \times (2\delta+1)$ gray differences of G were calculated by formula (2). The location $p_c(x_c, y_c)$ with the maximum G of the $(2\delta+1) \times (2\delta+1)$ iris center candidate points was considered as the final iris center.

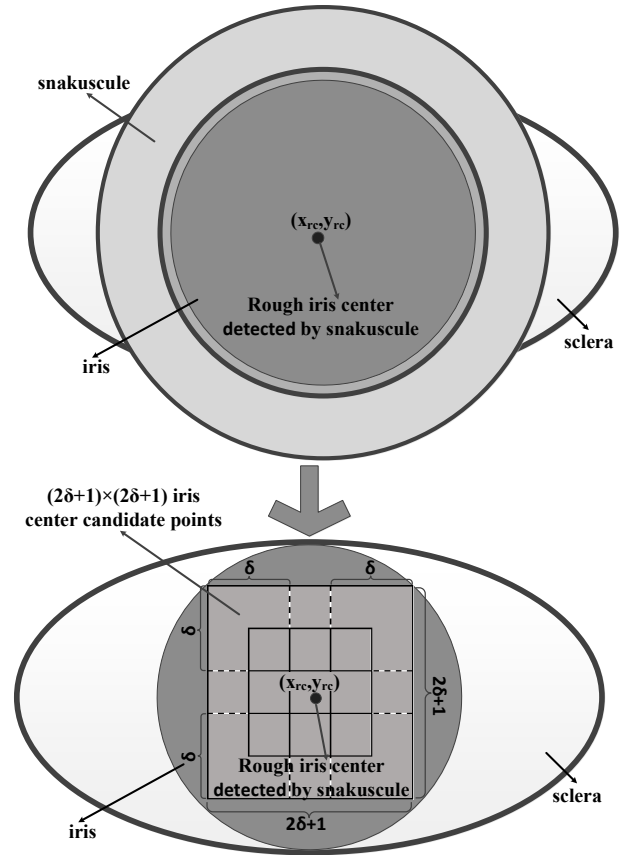


Figure 5. $(2\delta+1) \times (2\delta+1)$ iris center candidate points determined by the rough iris center (x_{rc}, y_{rc})

Anchor point

Anchor point is used as a reference point to compute the eye vector. The use of the inner or outer eye corners as the anchor points is the common approach (Valenti et al., 2009; Sesma et al., 2012) among regression-based methods for gaze tracking under natural light. However, Sesma et al. (2012) showed that the eye corners vibrate with eye rotations, which introduces errors in the gaze tracking. In Skodras et al. (2015), the anchor point was set as the center of the image patch tracked by the Lucas-Kanade inverse affine transform. However, sometimes eye corners or the center of the patch cannot be accurately tracked because they may be blocked or deformed in an image with a large head rotation. Therefore, a novel anchor point is designed as the reference point in this paper. The anchor point $p_a(x_a, y_a)$ is computed by

$$\begin{cases} x_a = \sum_{i=1}^n x_i / n \\ y_a = \sum_{i=1}^n y_i / n \end{cases} \quad (3)$$

where n is the number of facial landmarks and (x_i, y_i) is the coordinate of the i^{th} facial landmark.

There are two advantages to the anchor point. (1) The anchor point computed by stable facial landmarks does not vibrate with eye rotations and is not blocked with large head movements. (2) The mean of the facial landmarks can reduce the error compared with the single feature point near the eye area.

Head pose

The action of looking at objects usually involves head movement towards the object, and eye rotation focusing on the object. Fridman, Langhans, Lee, & Reimer (2016) used the head pose to track the gaze. However, Kennedy, Baxter, & Belpaeme (2015) found that gaze tracking merely based on the head pose is neither accurate nor consistent in human-robot interactions. Therefore, gaze tracking should synchronize eye rotation and head movement.

For the past several years, different methods for head pose estimation have been developed. The 2D-3D point correspondence methods achieve robust performance and can address large head movements. Therefore, the OpenCV iterative (Levenberg-Marquardt optimization) algorithm is used to estimate the head pose.

Mapping functions

After the eye vectors, head pose and screen coordinates have been obtained, the regression strategy is used to establish the mapping function between them. The linear terms, squared terms, cubic terms and interactions summarized in Blignaut (2016) are widely used for mapping eye vectors to screen coordinates. Unlike the head pose that was used to improve the eye vectors in Cheung and Peng (2015), it is directly introduced in the mapping functions. The mapping functions of n points with a polynomial of n or fewer terms can be expressed by

$$\begin{cases} g_h = \sum_{k=0}^{n-1} a_k e_h^i h_p^j h_y^l h_r^m, i, j, l, m \in [0, k] \\ g_v = \sum_{k=0}^{n-1} b_k e_v^i h_p^j h_y^l h_r^m, i, j, l, m \in [0, k] \end{cases} \quad (4)$$

where g_h and g_v are the POR of the horizontal and vertical directions, the coefficients a_k and b_k are determined by the calibration phase, e_h and e_v are the eye vectors of the horizontal and vertical directions, and h_p , h_y and h_r are the head pose angles of the pitch, yaw and roll, respectively. In this paper, six mapping functions derived by formula (4) are used to estimate the gaze. As shown in Table 2, the mapping functions of No.1 and No.2 use the linear and squared terms of eye vectors. No.3, No.4, No.5 and No.6 mix the linear and squared terms of eye vectors and head pose.

Table 2. Six mapping functions derived by formula (4), where the subscript "i" denotes "h" or "v".

No.	Mapping functions
1	$g_i = a_0 + a_1 * e_i$
2	$g_i = a_0 + a_1 * e_i + a_2 * e_i^2$
3	$g_i = a_0 + a_1 * e_i + a_2 * h_p + a_3 * h_y + a_4 * h_r$
4	$g_i = a_0 + a_1 * e_i + a_2 * e_i^2 + a_3 * h_p + a_4 * h_y + a_5 * h_r$
5	$g_i = a_0 + a_1 * e_i + a_2 * h_p + a_3 * h_y + a_4 * h_r + a_5 * h_p^2 + a_6 * h_y^2 + a_7 * h_r^2$
6	$g_i = a_0 + a_1 * e_i + a_2 * e_i^2 + a_3 * h_p + a_4 * h_y + a_5 * h_r + a_6 * h_p^2 + a_7 * h_y^2 + a_8 * h_r^2$

Ocular dominance theory is common and long-standing (Miles, 1930). In Nyström, Andersson, Holmqvist, & Van De Weijer (2013), a dominant eye is shown to be more accurate on SMI HiSpeed 500-Hz eye tracker systems. In addition, Quartley and Firth (2004) found that observers favor the left eye for leftward targets and the right eye for rightward targets. Furthermore, for relatively small eye-in-head rotations, Cui and Hondzinski (2006) proved that taking the average POR of the two eyes for gaze tracking is more accurate than using only one eye on remote eye tracker. In addition, one of the eyes may be blocked due to a large head movement. To unify these situations, a weight coefficient is used on the POR of the left and right eyes to revise the final POR of the horizontal g_{fh} and vertical g_{fv} directions.

$$\begin{cases} g_{fh} = w * g_{lh} + (1-w) * g_{rh}, w \in [0, 1] \\ g_{fv} = w * g_{lv} + (1-w) * g_{rv}, w \in [0, 1] \end{cases} \quad (5)$$

where w is the weight coefficient and g_{lh} , g_{rh} , g_{lv} and g_{rv} are POR of the horizontal and vertical directions of the left and right eyes, respectively.

Evaluation

Databases

The GI4E database (Villanueva et al., 2013) consists of 1236 images (800×600) from 103 different participants. Each participant has 12 images in which the participant gazed at different points on the screen. A large number of participants with low resolution images make it suitable for evaluating the performance of the proposed iris center localization method.

The EYEDIAP database contains RGB (640×480), RGB-D and HD (1920×1080) video clips from 16 participants. Continuous Screen (CS), Discrete Screen (DS) and 3D Floating Target (FT) are the stimuli that were used for the participants to gaze at. As shown in Figure 6, on the computer screen, DS target was drawn every 1.1 seconds on random locations and CS target was programmed to move along a random trajectory for 2s. The participants were asked to keep an approximately Static (S) or perform head Movements (M) when they gazed at the visual target. Each participant was recorded for 2 to 3 minutes. The proposed method was implemented on the RGB video clips that contains Discrete Screen with Static (DSS) and Discrete Screen with head Movements (DSM), and Continuous Screen with Static (CSS) and Continuous Screen with head Movements (CSM).

On the EYEDIAP database, the frame-by-frame screen target coordinates, head pose tracking states and eyes tracking states including the eyeballs' 3D coordinates have been provided in the files of "screen_coordinates.txt", "head_pose.txt" and "eyes

tracking.txt", respectively. It is noted that, a total of 52 RGB video clips of 13 participants were used to estimate the gaze in this paper because the 12th and 13th participants only recorded the video clips for 3D FT and the 7th participant's facial landmarks can be tracked on a small fraction of the entire RGB video clips due to the poor contrast.

Evaluation of iris center localization

The computation of the estimated eye center normalized error by use of formula (6) is suggested in Jesorsky, Kirchberg, & Frischholz (2001).

$$e = \frac{\max(d_{left}, d_{right})}{d} \quad (6)$$

where d_{left} and d_{right} are the distances between the estimated and labelled iris centers of the left and right eyes, and d is the distance between the labeled left and right iris centers. The estimated eye centers in the range of the normalized error $e \leq 0.05$ that are equivalent to locate in the pupil can be used for gaze tracking applications (Timm & Barth, 2011). Therefore, $e \leq 0.05$ is used as the benchmark to evaluate the optimal parameters of the iris center localization method in this paper.

For iris center localization, α , β and δ with different values were used on the GI4E database. The optimal values can be obtained when the number of eyes with a normalized error $e \leq 0.05$ reaches the maximum value. Therefore, values for α from 0.21 to 0.25 with the minimum interval of 0.05, β from 1.32 to 1.52 with the minimum interval of 0.04 and δ from 1 to 4 with the minimum interval of 1 were assessed in this paper

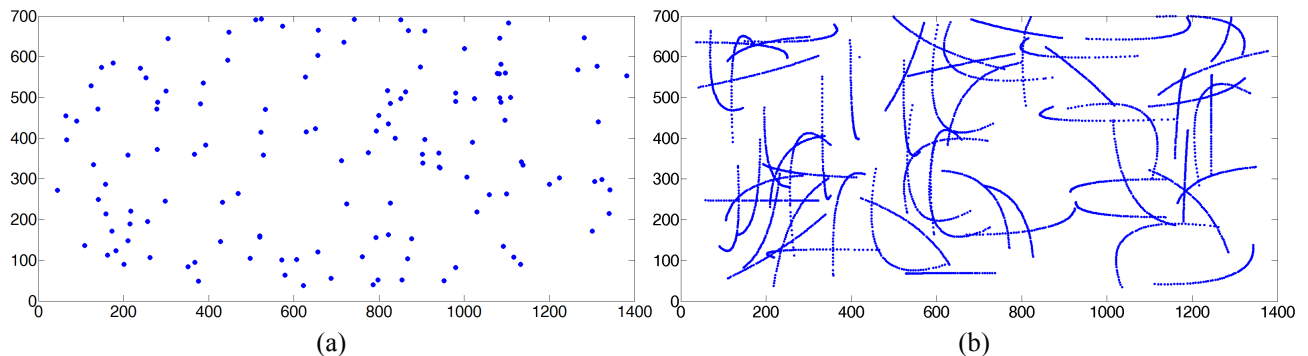


Figure 6. Example of screen coordinates for a video clips using (a) Discrete Screen target, (b) Continuous Screen target on the EYEDIAP database.

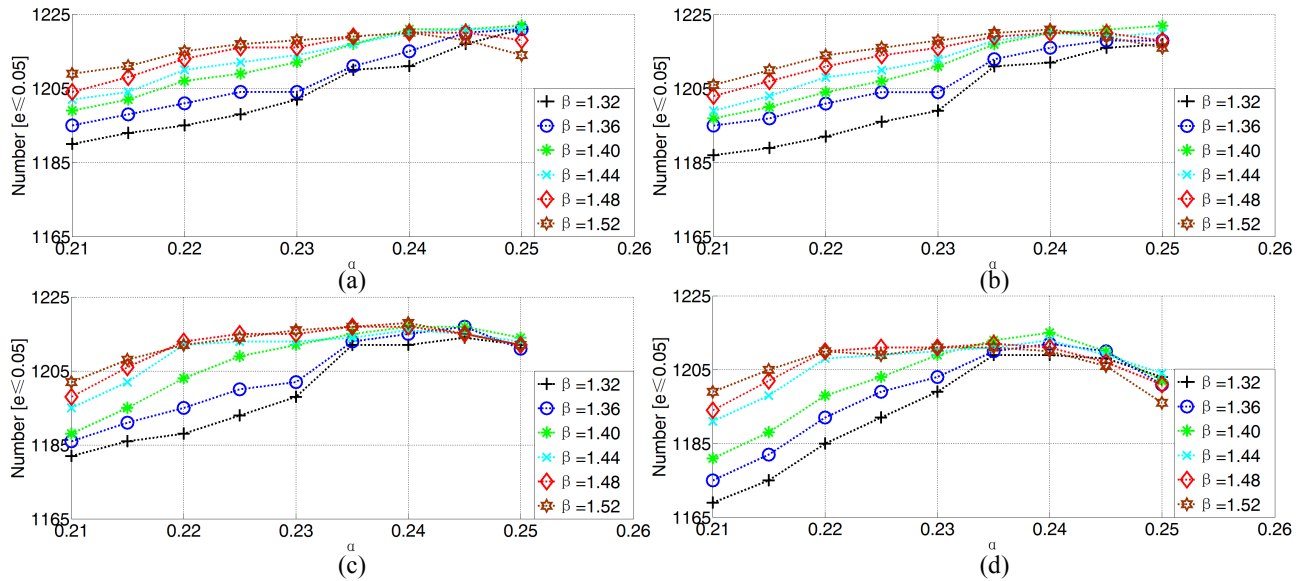


Figure 7. The number of images from the GI4E database with a normalized error $e \leq 0.05$ for different values of α , β and δ , where (a) $\delta = 1$ (b) $\delta = 2$ (c) $\delta = 3$ and (d) $\delta = 4$.

As shown in Figure 7, the maximum number of images with a normalized error $e \leq 0.05$ is 1222 when $\alpha = 0.25$, $\beta = 1.4$ and $\delta = 1$ or 2.

Evaluation of different mapping functions

The performance of different mapping functions was compared on the DSS, CSS, DSM and CSM RGB video clips. Iris centers were detected by the optimal parameters of $\alpha = 0.25$, $\beta = 1.4$ and $\delta = 1$ or 2. The anchor point was computed by formula (3), where the unstable facial landmarks around the mouth and eye areas were removed. Therefore, the parameter n equals 36. Then, the eye vectors for the horizontal and vertical directions were computed by the iris centers and the anchor point. Head pose was estimated by the iterative with six points (9, 31, 37, 46, 49 and 55 in Figure 2). Finally, the six mapping functions listed in Table 2 were used to estimate the gaze from the eye vectors and the head pose.

For each RGB video clip on the EYEDIAP database, the first 1000 frames that the faces could be detected were used as calibration frames, and the remaining frames were used as testing frames. The gaze tracking errors for the 13 participants were computed by averaging the results of the participants' testing frames. The gaze tracking error of each frame is computed by the POR of the left eye, the original 3D coordinate of the eye gaze screen point and the original 3D coordinate of the left eyeball.

The average gaze tracking errors of 52 RGB video clips that were used to evaluate the optimal mapping functions are shown in Table 3. Overall, the mapping functions of No.4 and No.2 achieved the best results in the horizontal and vertical directions, respectively. In addition, the gaze tracking errors show that $\delta = 2$ performs better than $\delta = 1$. Therefore, the following experimental results that involve the iris centers localization are conducted with $\delta = 2$. Meanwhile, the horizontal and vertical gaze tracking errors in the following experiments are regressed by the mapping functions of No.4 and No.2, respectively.

Table 3. The average gaze tracking errors (degrees) of 52 RGB video clips on the EYEDIAP database computed by six mapping functions.

No.	$\delta = 1$			$\delta = 2$		
	H	V	C	H	V	C
1	6.4	4.0	7.5	6.3	3.9	7.4
2	6.4	3.9	7.5	6.2	3.8	7.3
3	6.0	4.1	7.3	5.8	4.1	7.1
4	5.8	4.0	7.0	5.7	4.0	7.0
5	6.2	4.6	7.7	6.1	4.6	7.6
6	6.1	4.5	7.6	5.9	4.4	7.4

Note: No. denotes the mapping functions in Table 2. H, V and C are the horizontal, vertical and combined gaze tracking errors, respectively. C is the sum of the squares of the H and V. The minimum errors are marked as bold.

Evaluation of the weight coefficient w

In this paper, the weight coefficient $w \in [0,1]$ with a minimum value of 0.1 was used to compute the gaze tracking errors on the EYEDIAP database. As shown in Figure 8, $w = 0.5, 0.6$ and 0.5 achieve the best horizontal, vertical and combined gaze tracking errors, respectively. For simplicity, $w = 0.5$ was used in this paper.

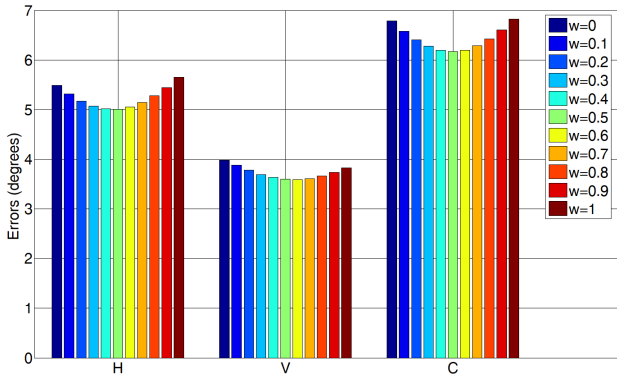


Figure 8. The H, V and C gaze tracking errors computed by different w on the EYEDIAP database.

To provide more comprehensive improved results by w for the 13 participants on the EYEDIAP database, the horizontal, vertical and combined gaze tracking errors on the DSS and CSS RGB video clips are shown in Figure 9.

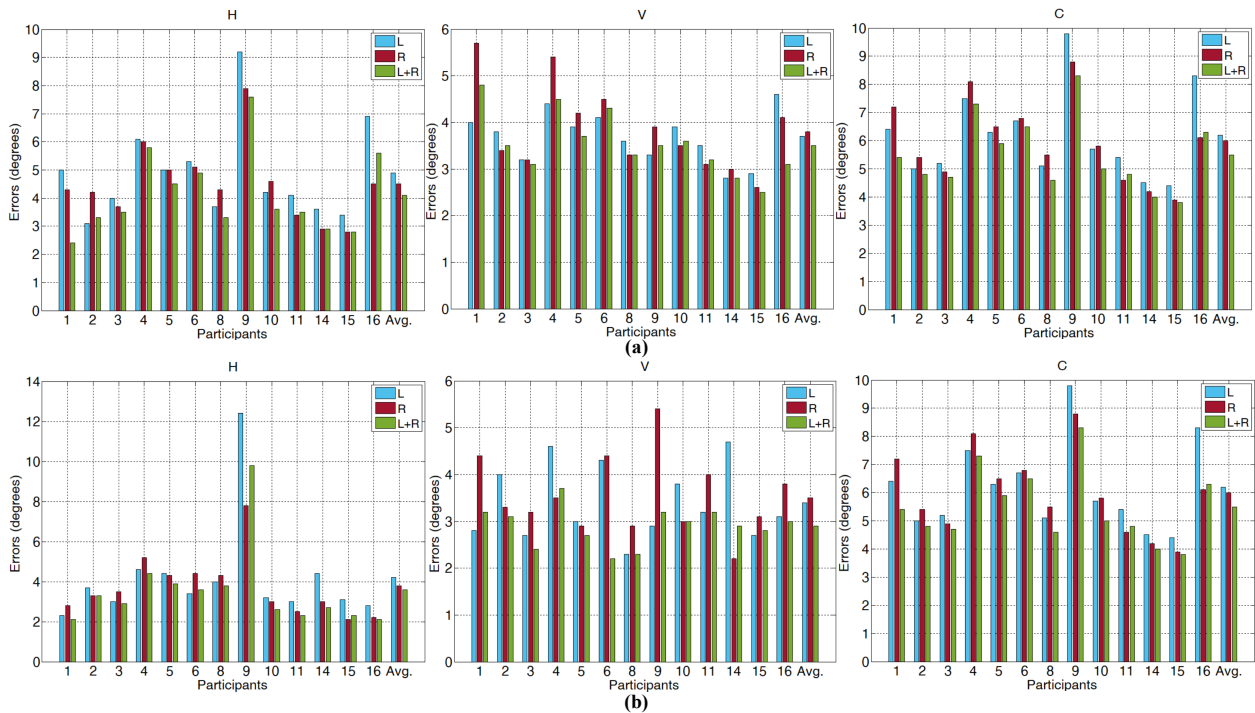


Figure 9. The H, V and C gaze tracking errors on the EYEDIAP (a) DSS and (b) CSS RGB video clips.

The results on the DSM and CSM RGB video clips are shown in Table 4. L ($w = 1$), R ($w = 0$) and L+R ($w = 0.5$) are the gaze tracking error computed by the POR of the left, right eyes and improved, the original 3D coordinate of the eye gaze screen point and the original 3D coordinate of the left and right eyeball. Meanwhile, the Total Frames (TF) and the Detected Frames (DF) are presented in the table. DF is a frame in which the face can be detected. Compared with the average face detection rate (DF/TF) of 86.9% in J.Li and S.Li (2016) on the CSM RGB video clips, an average of 97.2% was obtained in this paper. Considering the low quality frames and large head pose variations in the video clips, we believe the face detection rate is robust. As shown in Figure 9 and Table 4, most of the single eye gaze tracking errors are improved by averaging the POR of both eyes. The method achieved the average combined gaze tracking errors $5.5^\circ, 4.6^\circ, 7.2^\circ$ and 7.6° on the DSS, CSS, DSM and CSM RGB video clips. Compared with the gaze tracking error of 2.9° under natural light in Skodras et al. (2015) on their self-built database, the errors are high. The reason the errors are high is because that the EYEDIAP database has the lower quality eye image (the iris radius ≈ 4.5 pixels) compared to Skodras et al. (2015) self-built database (the iris radius ≈ 9 pixels).

Table 4. The gaze tracking errors (degrees) on the EYEDIAP CSM and DSM RGB video clips, where No. is the participant of the EYEDIAP database.

No.	DSM											
				H			V			C		
	TF	DF	DF/TF	L	R	L+R	L	R	L+R	L	R	L+R
1	4465	3893	87.2%	7.9	8.0	7.7	4.5	4.5	4.5	9.1	9.2	8.9
2	4464	4335	97.1%	7.6	6.5	6.8	3.7	3.8	3.7	8.4	7.6	7.7
3	4433	4322	97.5%	5.9	6.9	5.9	5.7	5.5	5.3	8.2	8.8	7.9
4	4464	4402	98.6%	7.7	7.2	5.7	5.0	4.8	4.5	9.2	8.7	7.3
5	4465	4465	100%	5.9	5.2	4.5	4.5	4.1	4.0	7.4	6.7	6.0
6	4464	4464	100%	5.3	5.5	5.1	4.6	4.6	4.5	7.0	7.2	6.8
8	4465	4020	90.0%	9.2	10.1	8.7	4.8	4.5	4.4	10.4	11.0	9.8
9	4464	4362	97.7%	10.1	6.8	7.5	3.7	3.7	3.6	10.8	7.8	8.3
10	4464	4450	99.7%	9.1	9.5	9.2	4.5	4.8	4.6	10.2	10.7	10.3
11	4465	4465	100%	4.7	3.7	3.5	4.5	6.4	3.5	6.5	7.4	4.9
14	4465	4464	100%	4.7	4.3	3.6	3.7	3.6	3.3	6.0	5.5	4.9
15	4465	4465	100%	3.8	3.4	3.2	4.1	4.2	4.2	5.6	5.4	5.3
16	4465	4286	96.0%	6.1	6.6	4.9	4.0	4.1	3.9	7.3	7.8	6.2
Avg.	4462	4338	97.2%	6.8	6.5	5.9	4.4	4.5	4.2	8.1	7.9	7.2
No.	CSM											
				H			V			C		
	TF	DF	DF/TF	L	R	L+R	L	R	L+R	L	R	L+R
1	4457	3370	75.6%	9.7	11.2	10.3	4.0	4.1	4.0	10.5	12.0	11.0
2	4457	4360	97.8%	7.9	7.6	7.6	3.2	3.2	3.1	8.5	8.2	8.2
3	4458	3962	88.9%	6.0	7.0	5.9	4.1	3.8	4.0	7.3	8.0	7.1
4	4494	4333	96.4%	8.1	7.0	6.7	3.7	3.8	3.7	8.9	8.0	7.7
5	4458	4394	98.6%	5.3	6.1	5.1	3.8	3.6	3.7	6.5	7.1	6.3
6	4458	4458	100%	7.7	8.4	7.6	4.4	4.5	4.1	8.8	9.6	8.6
8	4458	3510	78.7%	10.6	9.3	9.6	4.1	4.5	4.3	11.4	10.3	10.5
9	4457	4199	94.2%	7.4	7.2	7.2	4.0	3.8	3.8	8.4	8.1	8.1
10	4492	4492	100%	6.5	7.3	6.6	5.0	4.9	4.9	8.2	8.8	8.3
11	4458	4360	97.8%	6.0	6.5	6.2	3.6	3.6	3.6	7.0	7.4	7.1
14	4458	4439	99.6%	4.1	3.3	3.4	3.1	4.1	3.4	5.2	5.2	4.9
15	4458	4458	100%	3.6	3.5	2.9	3.2	3.2	3.1	4.8	4.8	4.2
16	4458	4293	96.3%	5.5	9.7	5.5	4.2	5.8	3.8	6.9	11.3	6.6
Avg.	4463	4202	94.2%	6.8	7.2	6.5	3.9	4.1	3.8	7.9	8.4	7.6

Computational cost

The method was realized by using the C++ language with Microsoft Visual Studio 2017, OpenCV and the dlib (King, 2009) library on a laptop with a 2.7-GHz Intel(R) Core(TM) i7-7500 processor and 8-GB RAM. Data from the EYEDIAP database and the laptop camera were used to measure the execution time, which was computed by averaging the processing time of all testing frames. The execution times of the proposed method are shown in Table 5. It is noted that facial landmarks detection includes face detection and landmarks detection. Experiment results show that face detection consumes most of

the processing time. Therefore, the original resolution (640×480) was resized to improve the efficiency of face detection. Facial landmarks are tracked on the faces from the raw frames, in which the faces are obtained by use of the rectangular face ROI detected in the resized frames. Unfortunately, the face detection rate of the EYEDIAP database decreased when the resolution is lower than 512×380 because the RGB video clips have small faces. Therefore, the execution speed is 22 fps for the EYEDIAP database. The mode of data from the laptop camera is closer to the practical system, which had an execution speed of 35 fps. Compared to the IR tracker with an execution speed in excess of 100 fps, natural light

trackers still have a long way to go to be usable in practice.

Table 5. The execution time of the gaze tracking system.

Data	Resolution	Execution time (milliseconds)		fps
		Facial landmarks detection	Gaze tracking	
EYEDIAP	512×380	44.5	0.7	22
Camera	320×240	27.4	1.2	35

Discussion

This paper aims to provide a gaze tracking system with a single-camera under natural light to extend its generality. The accuracies of the iris centers and the usability of the anchor point result in more applicable eye vectors. Using the eye vectors and the estimated head pose, second-order polynomial mapping functions are used to compute the POR of the horizontal and vertical directions on the screen. By implementing a weight coefficient on the POR of the left and right eyes, the final gaze errors improved. The iris center localization method has been shown to be accurate on the GI4E database, which consists of low resolution images under realistic conditions of 103 participants. With a normalized error $e \leq 0.05$, the feature position of the iris center has achieved an error as low as 1.13%.

Compared with the accuracy of 93.92% in Villanueva et al. (2013), the proposed iris center localization method presents a more accurate result of 98.87% for the feature position of the iris center. Moreover, it also outperforms all previous iris center localization methods in the same database. Compared with the average combined gaze tracking errors of 7.2° and 8.9° on the EYEDIAP CSS and CSM RGB video clips in Ghiass and Arandjelovic (2016), the proposed gaze tracking method reduced the errors by 36% and 14.6%, respectively. Compared with the average gaze tracking errors of 7.6° and 6.7° in horizontal and vertical directions on the EYEDIAP CSM RGB video clips in J.Li and S.Li (2016), 1.1° and 2.9°, respectively were reduced by the proposed method. Furthermore, the RGB and RGB-D video clips both were used as inputs in (Ghiass & Arandjelovic, 2016; J.Li & S.Li, 2016).

The gaze tracking errors are significantly better than the appearance-based and the model-based methods, indicating the effectiveness of the regression-based gaze

tracking method in low quality images. However, limited by the random gaze trajectories/points on the screen of which the EYEDIAP database is built, 1000 detected frames from the RGB video clips are used in the calibration phase. It is equivalent to the use of approximately 34 seconds from the 2 or 3 minutes of RGB video clips. In a practical application system, the calibration time could be reduced by calibration strategies summarized in Skodras et al. (2015) and the gaze tracking errors could be reduced by post-calibration regression in Blignaut, Holmqvist, Nyström, & Dewhurst (2014). In addition, considering the average gaze tracking errors shown in Table 3, the introduction of head pose in the mapping functions does not improve the accuracy of the vertical direction, but reduces the errors of the horizontal direction. The reason is that the eye vectors derived by the iris centers and the anchor point already contain some information of the head pose.

Although the algorithm in Kazemi and Sullivan (2014) presents robustness and accuracy, facial landmarks still cannot be tracked in some images especially on the CSM and DSM RGB video clips. Hence, in future work, the facial landmarks' algorithm should be improved in low quality images with large head movements. From the results in Figure 9 and Table 4, most of the single eye gaze tracking errors are improved by averaging the POR of both eyes. However, when one eye is blocked due to a large head movement, the w of the blocked eye should be decreased or set to 0. Meanwhile, w may be affected by the dominant eye that changes with the direction of the gaze (Quartley & Firth, 2004). Therefore, in the future, a dedicated database with large head pose variations, and various directions of gaze can be built to study choosing a better value of w . In addition, the mapping functions are regressed by person-special eye vectors, which results in a person-dependent gaze tracking system. A person-independent gaze tracking system can be researched by normalizing different people's eye vectors in a feature space.

A gaze tracking method with a non-intrusive sensor under natural light renders the system suitable for universal use on smartphones, laptops or tablets with a camera. The system, with an accuracy of approximately 6°, can be used in secure authentication of biometrics (Boehm et al., 2013) and gaze-based password entry fields for reducing shoulder-surfing (Kumar, Garfinkel, Boneh, & Winograd,

2007). The proposed gaze tracking method further bridges the interaction gap between humans and machines.

Ethics and Conflict of Interest

The author(s) declare(s) that the contents of the article are in agreement with the ethics described in <http://biblio.unibe.ch/portale/elibrary/BOP/jemr/ethics.html> and that there is no conflict of interest regarding the publication of this paper.

References

- Blignaut, P. (2016). Idiosyncratic Feature-Based Gaze Mapping. *Journal of Eye Movement Research*, 9(3). doi: 10.16910/jemr.9.3.2
- Blignaut, P., Holmqvist, K., Nyström, M., & Dewhurst, R. (2014). Improving the accuracy of video-based eye tracking in real time through post-calibration regression. In *Current Trends in Eye Tracking Research* (pp. 77-100). Springer International Publishing. doi: 10.1007/978-3-319-02868-2_5
- Boehm, A., Chen, D., Frank, M., Huang, L., Kuo, C., Lolic, T., & Martinovic, I. (2013, June). Safe: Secure authentication with face and eyes. *IEEE International Conference on In Privacy and Security in Mobile Systems (PRISMS)*. (pp. 1-8). Atlantic City, NJ, USA. doi: 10.1109/PRISMS.2013.6927175
- Cheung, Y. M., & Peng, Q. (2015). Eye gaze tracking with a web camera in a desktop environment. *IEEE Transactions on Human-Machine Systems*, 45(4), 419-430. doi: 10.1109/THMS.2015.2400442
- Crane, H. D., & Steele, C. M. (1985). Generation-V dual-Purkinje-image eyetracker. *Applied Optics*, 24(4), 527-537. doi: 10.1364/AO.24.000527
- Cui, Y., & Hondzinski, J. M. (2006). Gaze tracking accuracy in humans: Two eyes are better than one. *Neuroscience letters*, 396(3), 257-262. doi: 10.1016/j.neulet.2005.11.071
- Deubel, H., & Schneider, W. X. (1996). Saccade target selection and object recognition: Evidence for a common attentional mechanism. *Vision research*, 36(12), 1827-1837. doi: 10.1016/0042-6989(95)00294-4
- Drewes, J., Zhu, W., Hu, Y., & Hu, X. (2014). Smaller is better: Drift in gaze measurements due to pupil dynamics. *PLoS one*, 9(10), e111197. doi: 10.1371/journal.pone.0111197
- El Hafi, L., Ding, M., Takamatsu, J., & Ogasawara, T. (2017). Gaze Tracking and Object Recognition from Eye Images. *IEEE International Conference on Robotic Computing (IRC)* (pp. 310-315). Taichung, Taiwan. doi: 10.1109/IRC.2017.44
- Fridman, L., Langhans, P., Lee, J., & Reimer, B. (2016). Driver Gaze Region Estimation without Use of Eye Movement. *IEEE Intelligent Systems*, 31(3), 49-56. doi: 10.1109/MIS.2016.47
- Ghiass, R. S., & Arandjelovic, O. (2016, July). Highly accurate gaze estimation using a consumer RGB-D sensor. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence* (pp. 3368-3374). AAAI Press, New York, USA. URL: <https://dl.acm.org/citation.cfm?id=3061092>
- Hansen, D. W., & Ji, Q. (2010). In the eye of the beholder: A survey of models for eyes and gaze. *IEEE transactions on pattern analysis and machine intelligence*, 32(3), 478-500. doi: 10.1109/TPAMI.2009.30
- Holmqvist, K., & Andersson, R. (2017). *Eye tracking: A comprehensive guide to methods, paradigms, and measures*. Lund, Sweden: Lund Eye-Tracking Research Institute.
- Huck, A., Thompson, R. L., Cruice, M., & Marshall, J. (2017). Effects of word frequency and contextual predictability on sentence reading in aphasia: an eye movement analysis. *Aphasiology*, 31(11), 1307-1332. doi: 10.1080/02687038.2017.1278741
- Jesorsky, O., Kirchberg, K. J., & Frischholz, R. W. (2001, June). Robust face detection using the hausdorff distance. *International Conference on Audio-and Video-Based Biometric Person Authentication* (pp. 90-95). Springer, Berlin, Heidelberg. doi: 10.1007/3-540-45344-X_14
- Kaehler, A., & Bradski, G. (2016). *Learning OpenCV 3: computer vision in C++ with the OpenCV library*. "O'Reilly Media, Inc."
- Kazemi, V., & Sullivan, J. (2014). One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1867-1874). Columbus, OH, USA. doi: 10.1109/CVPR.2014.241

- Kennedy, J., Baxter, P., & Belpaeme, T. (2015, March). Head pose estimation is an inadequate replacement for eye gaze in child-robot interaction. In Proceedings of the tenth annual acm/ieee international conference on human-robot interaction extended abstracts (pp. 35-36). ACM. New York, NY, USA. doi: 10.1145/2701973.2701988
- Kim, K. N., & Ramakrishna, R. S. (1999). Vision-based eye-gaze tracking for human computer interface. IEEE SMC'99 Conference on Systems, Man, and Cybernetics Proceedings. (Vol. 2, pp. 324-329). Tokyo, Japan. doi: 10.1109/ICSMC.1999.825279
- King, D. E. (2009) Dlib-ml: a machine learning toolkit. *J Mach Learn Res* 10(Jul):1755–1758.
- Krafka, K., Khosla, A., Kellnhöfer, P., Kannan, H., Bhandarkar, S., & Matusik, W. (2016). Eye tracking for everyone. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2176-2184). Las Vegas, NV, USA. doi: 10.1109/CVPR.2016.239
- Kumar, M., Garfinkel, T., Boneh, D., & Winograd, T. (2007, July). Reducing shoulder-surfing by using gaze-based password entry. In Proceedings of the 3rd symposium on Usable privacy and security (pp. 13-19). ACM. Pittsburgh, Pennsylvania, USA. doi: 10.1145/1280680.1280683
- Lahey, J. N., & Oxley, D. (2016). The power of eye tracking in economics experiments. *American Economic Review*, 106(5), 309-13. doi: 10.1257/aer.p20161009
- Li, J., & Li, S. (2016). Gaze estimation from color image based on the eye model with known head pose. *IEEE Transactions on Human-Machine Systems*, 46(3), 414-423. doi: 10.1109/THMS.2015.2477507
- Miles, W. R. (1930). Ocular dominance in human adults. *The journal of general psychology*, 3(3), 412-430. doi: 10.1080/00221309.1930.9918218
- Mora, K. A. F., Monay, F., & Odobez, J. M. (2014, March). Eyediap: A database for the development and evaluation of gaze estimation algorithms from rgb and rgb-d cameras. In Proceedings of the Symposium on Eye Tracking Research and Applications (pp. 255-258). ACM. doi: 10.1145/2578153.2578190
- Mora, K. A. F., & Odobez, J. M. (2012, June). Gaze estimation from multimodal kinect data. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (pp. 25-30). Providence, RI, USA. doi: 10.1109/CVPRW.2012.6239182
- Newman, R., Matsumoto, Y., Rougeaux, S., & Zelinsky, A. (2000). Real-time stereo tracking for head pose and gaze estimation. *Fourth IEEE International Conference on Automatic Face and Gesture Recognition* (pp. 122-128). Grenoble, France. doi: 10.1109/AFGR.2000.840622
- Nyström, M., Andersson, R., Holmqvist, K., & Van De Weijer, J. (2013). The influence of calibration method and eye physiology on eyetracking data quality. *Behavior research methods*, 45(1), 272-288. doi: 10.3758/s13428-012-0247-4
- Pan, Y., & Steed, A. (2014). A gaze-preserving situated multiview telepresence system. In Proceedings of the 32nd annual ACM conference on Human factors in computing systems (pp. 2173-2176). ACM. Toronto, Ontario, Canada. doi: 10.1145/2556288.2557320
- Quartley, J., & Firth, A. Y. (2004). Binocular sighting ocular dominance changes with different angles of horizontal gaze. *Binocular vision & strabismus quarterly*, 19(1), 25-30.
- Sesma, L., Villanueva, A., & Cabeza, R. (2012). Evaluation of pupil center-eye corner vector for gaze estimation using a web cam. In Proceedings of the symposium on eye tracking research and applications (pp. 217-220). ACM. doi: 10.1145/2168556.2168598
- Skodras, E., Kanas, V. G., & Fakotakis, N. (2015). On visual gaze tracking based on a single low cost camera. *Signal Processing: Image Communication*, 36, 29-42. doi: 10.1016/j.image.2015.05.007
- Sugano, Y., Matsushita, Y., & Sato, Y. (2014). Learning-by-synthesis for appearance-based 3d gaze estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1821-1828). Columbus, OH, USA. doi: 10.1109/CVPR.2014.235
- Thevenaz, P., & Unser, M. (2008). Snakuscles. *IEEE Transactions on image processing*, 17(4), 585-593. doi: 10.1109/TIP.2007.914742

- Timm, F., & Barth, E. (2011). Accurate Eye Centre Localisation by Means of Gradients. VISAPP 2011 - Proceedings of the Sixth International Conference on Computer Vision Theory and Applications, 125-130. URL: <https://pdfs.semanticscholar.org/c931/1a0c5045d86a617bd05a5cc269f44e81508d.pdf>
- Valenti, R., Staiano, J., Sebe, N., & Gevers, T. (2009). Webcam-based visual gaze estimation. Image Analysis and Processing-ICIAP, 662-671. doi: 10.1007/978-3-642-04146-4_71
- Villanueva, A., Ponz, V., Sesma-Sanchez, L., Ariz, M., Porta, S., & Cabeza, R. (2013). Hybrid method based on topography for robust detection of iris center and eye corners. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 9(4), 25. doi: 10.1145/2501643.2501647
- Weidenbacher, U., Layher, G., Strauss, P. M., & Neumann, H. (2007, September). A comprehensive head pose and gaze database. 3rd IET International Conference on Intelligent Environments (IE 07), 455-458. Ulm, Germany. doi: 10.1049/cp:20070407
- Wood, E., & Bulling, A. (2014, March). Eyetab: Model-based gaze estimation on unmodified tablet computers. In Proceedings of the Symposium on Eye Tracking Research and Applications (pp. 207-210). ACM. doi: 10.1145/2578153.2578185
- Xiao, F., Huang, K., Qiu, Y., & Shen, H. (2018). Accurate iris center localization method using facial landmark, snakuscule, circle fitting and binary connected component. Multimedia Tools and Applications, 1-21. doi: 10.1007/s11042-018-5787-x