Eye-wearable head-mounted tracking and gaze estimation interactive machine system for human–machine interface

Ko-Feng Lee1, Yen-Lin Chen1, Chao-Wei Yu1, Cheng-Lung Jen2, Kai-Yi Chin3, Chen-Wei Hung4 and Chih-Bo Wen1

Abstract
In this study, a head-mounted camera was used to track eye behaviors and estimate the gaze point on the user’s visual plane. The integration of the elastic mechanism design makes the headset adaptable for various users. The wearable cases were prototyped with low-cost cameras to produce an efficient eye tracking solution. This proposed system can effectively extract and estimate pupil ellipse from a few camera images of an eye and compute the corresponding three-dimensional eye model. The system can match later images of the same pupil ellipse from a head-mounted camera to give the possible visual angles. To estimate the gaze point, the system uses multiple-point calibration to solve the related polynomial formula for future angle-to-gaze mapping. The proposed eye-tracking algorithms can provide a low-complexity solution with high accuracy, precision, and speed. This tracking system is a low-cost and promising system that can be used in headsets for virtual reality, auxiliary equipment, interactive machine, and human–machine interface applications. The proposed eye-tracking algorithm can achieve satisfactory performance without using a high-end high-speed camera and can be detected under different lighting sources, and the average errors of the detection results are stably within 9 pixels and at a distance of 50 cm from the screen; while the average error of the fixation mapping results is within 3°.

Keywords
Eye tracking, gaze estimation, machine vision, head-mounted, human–machine interface

Introduction
Human–machine interface (HMI) is a field covers multiple technologies including computer science, artificial intelligence, sociology, etc. The definition of HMI is that when person and machine communication by some kind of interface. Person can control the machine and the machine can report status to the user through the interface. In the other hand, the correct communication of messages and control between humans and computers is the main definition of the HMI. For example, Jira et al.1 developed a gesture-based human–machine interface for vehicle driving using finger pointers that provide user for controlling an in-vehicle system.

As mentioned above, how to make the user control the machine is an important issue. There are many ways to let the user control the machine. The most traditional is to control by using program instructions, but it is also the
most difficult way. Therefore, there are many easier ways to implement control the machine. The easiest and least laborious way is to use visual control.

Vision is crucial to human cognitive behavior. In humans, eyes are one of the primary sensory organs and can distinguish between features of light, distance, and color. The central fossa of the eyeball is densely populated with photoreceptor cells, making it the most sensitive part of the visual perception zone. Eye movements enable the gaze to focus on the central fossa and thereby obtain clear visual information. Therefore, several studies raised the issues of using visual information for HMI and interactive machine.

As mentioned, eye movements are closely related to the target under observation. In addition, the trajectory of gaze probably represents cognitive vision and processing history. In daily life, people look at the things that they have a natural desire to interact with. It is even possible to analyze the state of mental activity through the trajectories of eye movements.

An eye tracker, also known as an eye movement tracking instrument, is an instrument that is used to record eye movements and convert them into gaze trajectories. In recent years, practical products with integrated eye trackers have been commercially produced. The results of gaze tracking can be used for related research in neuroscience, psychology, education, marketing, and advertising analysis.

In addition to being used for research, the eye tracker is also a communication aid for physically disabled people that can be used to express self-awareness through behaviors such as eye movements and blinking. Therefore, for patients with problems such as motor neuron disease, muscular dystrophy, cerebral palsy, spinal injury, multiple sclerosis, or stroke or for those with difficulty speaking, the eye tracker can be used as an aid for communication with the outside world.

Since 1990, various eye trackers have been developed for specific applications; however, because such products are still expensive, they are usually used for research purposes. Therefore, this study proposed a low-cost wearable device that could be developed into a low-cost eye tracker.

In recent years, wearable devices have gradually emerged. Some head-mounted devices allow users to have immersive experiences in virtual reality. Such devices can enable the user to switch between scenes regardless of time and space and are widely used in games and education. Therefore, the gaze point and its movement trajectory can also provide a quick and intuitive secondary indicator for the human–machine interface and provide optimized information on the drawing operation to improve the user experience.

The most detection algorithms for eye trackers have been classified as model-based methods and feature-based methods. Model-based methods usually match the vision with preset shapes and model formulas and then determine the optimal solution among the limited candidates through voting or classification. With the correct model, the model-based method can consistently locate the iris. For a typical model-based method, the difficult part is the stable and accurate determination of the iris position on the basis of the reflection points created by the ambient light sources shining from different directions. Feature-based methods usually use the features of the eyes. A typical feature-based method fits the shape of the eye in a circular or elliptical shape to locate the center of the iris. Compared with model-based methods, feature-based methods may require a much smaller code that can perform in real time; however, a typical feature-based method is unfavorable compared with a typical model-based method in terms of stability and accuracy.

This study used a highly stable and accurate feature-based method to track gaze and adjusted a pupil elliptical extraction method to make it suitable for head-mounted applications to achieve the purpose of the HMI and develop an interactive machine. For the gaze-point estimation, the polynomial equations of the mapping model are extended to enable the three-dimensional fixation vector to correspond to the two-dimensional visual plane. The correction procedure of the eye model, which simplifies the calibration process of the overall system, minimizes the correction time, and improves ease of use, was integrated into the sampling period established by the mapping model.

Therefore, this study proposes a set of eyesight devices that can capture eyeball images through the Logitech Quick Cam Pro 5000 webcam to achieve the low-cost wearable devices.

**Related literature**

In the 20th century, Huey created the first invasive eye tracker, as displayed in Figure 1; the device resembled a contact lens with a hole positioned near the middle of the pupil. The remainder of the contact lens was connected to an aluminum pointer; eyeball and pointer movements were synchronized to capture the movement trajectory of the eyeball. However, this was uncomfortable for the subject. In addition, the subject was required to describe the
movement of the eyeball and compare it with the captured data. Thus, the results were easily influenced by subjective feelings.

As depicted in Figure 2, Babcock et al.\textsuperscript{7} created the first noninvasive eye tracker in 1935. This eye tracker sent a beam of light onto the subject's eye and then used photographic film to record the reflection from the eye. The continuous light spot recorded on the film was used as the moving trajectory of the eyeball, and the position of the gaze point was analyzed accordingly. The results of the study were more objective than were those of the invasive eye tracker. However, this noninvasive eye tracker was not popular because of its high cost.

The detection algorithms for eye trackers have been classified into model-based methods and feature-based methods. Model-based methods usually consider entire images in the context of preset shapes and model
formulas. For example, Kao et al.⁴ proposed a model-based eye tracker. Kao’s model-based method first defined an eye model and then calculated the angle of rotation of the model. The angle of rotation can match the position of the iris on the image with the position of the iris defined in the model. It can be seen that the center position of the iris on the image is the position of the iris in the model after being rotated through a certain angle. Swirski and Dodgson⁵ proposed that it is also a shape-based eye tracker, which uses a pupil image to extract an ellipse and back-projection ellipse into a stereo space. After multiple sampling operations, an eye model can be generated. The pupil ellipse that is subsequently captured can fit the most likely fixation angle from the model. It is computationally intensive and does not rely on reflected spots for tracking.

The feature-based approach uses the common characteristics of the human eye to identify the eye. These include the corneal edge, pupil, or the reflection of light on the cornea. The main purpose of each feature is to obtain information concerning the eyes and face. Therefore, it is not highly sensitive to changes in illumination. For example, Yang used a nonlinear filter to identify suitable candidate points for formalizing information of the iris. However, Yang et al. used method that required high-quality images for accurate detection.⁸ Yang et al.⁹ used the edge color information of the pupil and iris to obtain the threshold value from the color information around the eye. Sirohey et al. used iterative methods to determine the most suitable threshold value for distinguishing skin color using the model that establishes the eye frame. However, in environments with different brightness levels, the detection tended to fail due to the similarity of eyebrows to shadows.⁹ Sigut and Sidha proposed a method for supporting head movement to combine the bright spots reflected on the iris based on shapes and additional light sources. The distance from the center point of the iris to the point of reflection can be used to determine the distance and direction of head movement to correct the effect of head movement.¹⁰ Lee et al. proposed a method for a portable device that involved installing multiple sets of light-emitting diode lights on the device. These light sources reflected the spots on the iris; Lee et al. calculated the dependent positions of each of these points. The vector correction of head rotation was calculated to correct the fixation deviation caused by head movement.¹¹ Jen et al.¹² proposed a new wearable eye-gaze tracking system with a single webcam mounted on glasses; this system involves skin detection and eyelid removal to extract the region of interest. Dobeš et al.¹³ developed a successful method of eye and eyelid localization on a computer using a modified Hough transform. The efficiency of this method was tested on two publicly available face image databases and one private face image database.

The study in Kao et al.⁴ proposed a model-based detection method. This paper is a feature-based detection method. The comparative literature⁴ is also a feature-based detection method. This paper uses the Hough circle, the least-squares method, and the particle-filtering method to obtain the eye-tracking results.⁴ However, in the preprocessing part, the gray-scale morphology and Sigmoid function are used to increase the light and shadow. Changed resistance, and does not use the region of interest to limit the positions of iris detection. Under low lighting and strong lighting conditions, the range of the regions of interest will be cut into the eye, and the iris outside the region will be detected.

In Kao et al.,⁴ a high-end and high-speed camera is adopted for eye detection and tracking. This study can accurately be tracked using a low-cost consumer webcam. Comparably, the study in Kao et al.⁴ needed a chin to fix the human head position. The input images can be taken by a wearable device in a mobile environment. The advantages of this study are as follows:

1. The average error of the detection rate is stably within 9 pixels.
2. Under the 50-cm distance from the screen, the error of the fixation mapping is within 3°.
3. Feasibility under different light sources.
4. Accurate detection and tracking of iris center using a low-cost and low-speed camera.

This study combined a low-cost head-mounted device with a feature-based method. An image preprocessing method helped to enhance the characteristics of the iris edge and fit a circle with the characteristics of the iris edge when the system had located the center of the iris. The center of the circle represented the center of the iris and was tracked using a filter. This study developed a low-cost head-mounted eye tracker to provide a communication mechanism for people with disabilities.

**Device**

The device flowchart is depicted in Figure 3. When the system is operating, a Logitech Quick Cam Pro 5000 captures eyeball images at 30 fps with a resolution of 320 pixels × 240 pixels. The outer casing is removed, and part of a photographic lens is fixed on a pair of glasses. Our built-in visible eye tracker processes gaze points,
the coordinates of which are subsequently output and mapped onto the screen to present the eye movement trajectory.

When performing fixation and screen gaze-point mapping corrections, the user must remain at a fixed distance from the screen. As shown in Figure 4, a chin support is adopted to assist users in maintaining a fixed distance between the user’s head and the screen.

The photograph presented in Figure 5 indicates that errors are caused in the test environment as a result of distance and screen size; the optimal distance for testing is approximately 50 cm.

**Method**

The system architecture diagram is presented in Figure 6. This study had three steps. First, image preprocessing was used when the camera captured the image. After thorough image preprocessing, eye detection was conducted. Third, eye tracking and mapping were conducted.

**Preprocessing**

This study used gray morphology as the preprocessing to increase the amount of iris information obtained. After obtaining the iris information, this process first extracted the part of the iris in the image, then this
study uses automatic multilevel thresholding technique to achieve contrast enhancement. As a result, this study will observe the first 10 images. If the iris features are too rare, they will use to create the iris mask to achieve feature enhancement. Its details are as follows:

This study used grayscale images for preprocessing to increase the amount of iris information obtained. After iris information had been added, an automatic multilevel thresholding technique\textsuperscript{14–16} obtained the regions of interest for the iris in the image. Mathematical morphology operations of dilation and erosion were then conducted on the thresholded regions. This study used circular structural elements to conduct morphological operations with circular structural elements.

\begin{align*}
(f \ominus b)(x, y) &= \min_{(s, t) \in b} \{f(x + s, y + t)\} \\
(f \oplus b)(x, y) &= \max_{(s, t) \in b} \{f(x + s, y + t)\}
\end{align*}

\textbf{Figure 5.} Testing process.

\textbf{Figure 6.} System architecture diagram.
Formulas (1) and (2) describe morphological erosion and dilation, respectively. Formula (1) describes the erosion operation, where $f$ denotes grayscale images, $b$ denotes structural elements, $x$ and $y$ denote coordinates, and $s$ and $t$ denote variables within the structural elements. The system moves structural elements over the image and seeks the minimum value within the structural elements. Formula (2) describes the dilation operation, where $b$ denotes the corresponding structural elements. The structural elements scan the image and identify the maximum value within the scanned elements.

As illustrated in Figure 7, the morphological erosion operation magnifies the darker parts of the grayscale image. The dilation process enlarges the brighter part of the grayscale image. Grayscale images must be treated first, but after erosion, the iris is deformed in the image. When the system executes the dilation process, it attempts to return the iris to as close to its normal size as possible.

After the system has performed grayscale morphology, it subjects the images to automatic multilevel thresholding. However, the general automatic multilevel thresholding method has considerable difficulty with thresholds. Therefore, this study used an automatic multilevel thresholding algorithm to automatically find the threshold value. An automatic multilevel image thresholding system can homogenize a grayscale image as a histogram. Automatic multilevel thresholding can then be conducted to identify the most suitable thresholds for different images. To achieve optimal image characteristics, this study performed two runs of automatic multilevel thresholding on images to automatically determine the threshold. In this study, the first run of automatic multilevel thresholding with images tended to over amplify eyelid features; this obscured the characteristics of the iris. The second run of automatic multilevel thresholding with images dramatically increased the resolution of iris characteristics and reduced other noise to increase the iris detection rate. Histogram equalization served to redistribute and accumulate pixel values of grayscale images. As depicted in Figure 8, the quantity corresponding to the pixel value corresponded with the cumulative amount. Histogram homogenization was ineffective for redistribution. Therefore, this study used the sigmoid function to perform the redistribution.

**Locating the iris center**

After image preprocessing, the iris center must be located. Locating the position of the center of the iris requires two steps: the first uses the Hough circle, and the second uses the least-square circle-based method. Finally, the detected iris center must be tracked through the particle filter.
Figure 9 presents a diagram of the Hough circle detection process. The image subjected to the image feature enhancement process is used as the input for Hough circle detection. The system can obtain a circle after performing the Hough circle conversion. However, the reliability of detection using the Hough circle is insufficient; therefore, the least-square circle-based method must be used for correction.

The system cannot clearly define the pupil position using Hough circle detection in the image. Therefore, this study used the difference between the iris and the sclera, which is called the limbus, as the basis for judgment. If the Hough circle detection returns the wrong position for the center of the iris, it must be corrected using the least-square circle-based method. The least-square circle-based method\(^{19}\) scatters the center of the correct result to obtain the feature points of the heterochromatic edge. As shown in Figure 10, two steps must be executed to identify the limbus features: (a) The radius of the largest circle detected by the Hough circle is used as the scattering radius. The system extends \(N\) rays outward at an angle of \(240^\circ\). The eyelids mostly cover the upper edge of the eyeball. Therefore, the divergence angle is less than \(120^\circ\) above the upper edge of the image. (b) The system must stop the search for the ray when it encounters a black pixel juxtaposed with a white pixel. These points are the feature of the limbus. After obtaining the features of several different limbi, the system can use these features to find the most suitable circle.

Formula (3) is used for the least-square circle-based method; \(x\) and \(y\) are the coordinates of the feature points of the heterochromatic edges.

\[
x^2 + y^2 + ax + by + c = 0
\]

(3)

When the parameters \((a, b, c)\) are obtained from formula (3), as shown in formula (4), this minimizes the total error of all feature points and the circle. The center of the most suitable circle for representing the iris can be used as the center of the iris.

\[
e = \sum_{i=1}^{n} (x_i^2 + y_i^2 + ax_i + by_i + c)^2
\]

(4)

As indicated in formulas (5), (6), and (7), the three unknowns \(a, b,\) and \(c\) can be partially differentiated. The least error occurs when the partiality is 0.

\[
\sum_{i=1}^{n} 2(x_i^2 + y_i^2 + ax_i + by_i + c)x_i = 0
\]

(5)
As demonstrated in equations (8) and (9), after obtaining values for $a$, $b$, and $c$, the system can obtain the center coordinate $p(x, y)$ of the circle and the radius $R$

$$p(x, y) = \left(-\frac{a}{2}, -\frac{b}{2}\right)$$

$$R = \sqrt{\frac{a^2 + b^2 - 4c}{4}}$$

Figure 11 depicts the result of least-square circle-based detection, which provides the center of the circle, the radius of the center, the standard deviation of the radius of the feature points of the different color edges, and the ratio of the black and white pixels in the detection result circle. These items can be used for follow-up tracking in target feature alteration.

**Tracking the iris center**

The method of locating the iris center in motion pictures is difficult to use for detection. Therefore, this study added tracking methods to compensate for the lack of detection. For example, the most common tracking method is the use of the Kalman filter.\(^{20-22}\) The Kalman filter is used to make predictions, and then the results are corrected using the Gaussian distribution and a linear model. However, the Kalman filter employs linear tracking, whereas the movement of the eyeball is nonlinear. Therefore, this study used particle filters for tracking rather than the Kalman filter. Particle filters utilize nonlinear tracking and can design feature and weight values.

As illustrated in Figure 12, the original algorithm for the particle filter is divided into three stages: (1) random sampling; (2) importance sampling; and (3) resampling. First, the particle filter must determine whether tracking is being performed for the first time. If so, then the system must distribute the particles according to the results for the center of the iris. If tracking is not being performed for the first time, then the system must distribute the particles according to the results of the previous trace. Importance sampling constitutes the second stage and has enabled the accurate tracking of particle filters in many studies. Importance sampling serves to determine whether each particle is the same as the target being tracked and assign each particle a weight value. Calculations based on this feature produce the weight value. Finally, the particle with the highest weight value is used as the tracking result. The third phase, resampling, mainly serves to distribute the position of each particle in advance. The distribution is based on the weight value of each particle. Particles with high weight values can generate numerous new particles around them. Particles with low weight values can only exist and cannot produce new particles around them. This stage does not produce an image of the tracking results. Resampling redistributes the position of the new particle based on the weight. After the particles with high weight values have been
redistributed, new particles can be generated around them. Only particles with low weight values produce few or no new particles. The object is tracked back and forth until it has disappeared. Figure 13 displays the flow chart of tracking in this study.

Formula (10) describes random sampling. The system first confirms whether tracking is being performed for the first time when particles begin entering the particle filter. The first tracking randomly distributes the particles to locate the center of the iris, and subsequent tracking is employed to randomly distribute the particles on the basis of the previous tracking result. \(X_t\) is the target state at that moment, and \(X_{t-1}\) is the state noise of the previous tracking result. \(n_t\) is a Gaussian distribution of noise probability. The state of the tracking target is the same as the state of the previous target. However, variations in target state at different times may be caused by measurement error or changes in the state or light distribution of the target.

\[
X_t = X_{t-1} + n_t \tag{10}
\]

Formula (11) indicates that the input of the research system is a flat image. Therefore, the target position can be divided into a horizontal X-direction and vertical Y-direction. When using noise, the position of each moment should differ slightly. \(X_t\) and \(Y_t\) are the right-angle coordinates representing the target position at time \(t\). The coordinate position \((X_{t-1}, Y_{t-1})\) at this moment is the coordinate position of the previous moment plus
noise \( n_x \) and \( n_y \).

\[
\begin{bmatrix}
X_t \\
Y_t
\end{bmatrix} = \begin{bmatrix}
X_{t-1} \\
Y_{t-1}
\end{bmatrix} + \begin{bmatrix}
n_x \\
n_y
\end{bmatrix}
\]  

(11)

Formula (12) presents the Gaussian distribution. \( Q \) is a covariance matrix. \( \sigma_x \) is the standard deviation for horizontal distribution, and \( \sigma_y \) is the standard deviation for the vertical distribution. Formula (13) was obtained by analyzing the change in the position of the ground truths, which indicates the range of possible displacements of the target. This standard deviation changes in a dynamic manner when the eye moves violently.

\[
n_t \sim N(0, Q)
\]

(12)

\[
Q = \begin{bmatrix}
\sigma_x^2 & 0 \\
0 & \sigma_y^2
\end{bmatrix}
\]

(13)

This step replaces the resampled portion. Therefore, this study uses the tracking result coordinates at the previous moment and the result coordinates for the center of the iris at this moment as a reference to adjust the range in which the particle group can be dispersed.

\[
Z_t = g(X_t, v_t) = \begin{bmatrix}
p'_t \\
r'_t \\
e'_t \\
d'_t
\end{bmatrix}
\]

(14)

\[
r = \frac{1}{n} \sum_{i=1}^{n} \text{dist}(p_{\text{center}}, p_i)
\]

(15)

\[
e = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\text{dist}(p_{\text{center}}, p_i) - r)^2}
\]

(16)

\[
d = \frac{\sum_{p_{\text{in}} \in S} \text{black}(p_{\text{in}})}{|S|}
\]

(17)

Importance sampling must identify the features of each particle, compare the features of the tracking target, and then identify the weight value of each particle. A high weight value indicates that the target is large. A lower weight value indicates that the particle is less similar to the target. As indicated in formula (14), \( Z_t \) is the target feature vector; \( g \) is the observation function. \( X_t \) denotes the current real state of the target, \( v_t \) is a Gaussian probability model. \( p'_i = (x_i, y_i) \) is the location of particle \( i \) at a given moment, \( e'_i \) is the standard deviation of the radius produced by the iris edge feature points at that moment of fitting the circle, and \( r'_i \) is the radius of the circle fitted by the iris edge feature points \( n \) for this reference on \( p'_i \). As given in formula (15), \( p_{\text{center}} \) is the coordinate position of \( \text{Particle}_i \); \( r \) is the radius of the circle fitted by the \( p_{\text{center}} \). In formula (16), \( d'_i \) is the black pixel ratio in \( S = \{p_{\text{in}}|\text{dist}(p_{\text{in}}, p_{\text{center}}) < r\} \); \( e \) is the standard deviation of the radius of the iris edge of feature point \( p_i \) and the fitted circle. In formula (17), \( d \) is the proportion of the black pixels in the fitted circle.

\[
diff'_i = -|F_{t-1} - F'_t|
\]

(18)

\[
\frac{exp(diff'_i)}{\sum_{i=1}^{n} exp(diff'_i)}
\]

(19)
In formula (18), $\text{diff}_i$ represents the characteristic difference between particle $i$ at time $t$ and the tracking result obtained at time $t-1$; the difference between the feature of particle $i$ at time $t$ and the tracking result obtained at time $t-1$ is greater. The difference is calculated for time $t$. A negative number to reverse it must multiply the difference. The formula is used to calculate the difference between each particle sample and the tracking target.

In formula (19), $w_i^t$ represents the weight value of particle $i$ at time $t$. The formula can normalize the difference and be used to calculate the corresponding weight value. In formula (20), $c_i$ represents the distance between the coordinates of the previous tracking result and the coordinates of particle $i$; $r_i$ represents the difference between the radius of the previous tracking result and the that of particle $i$; $e_i$ represents the difference in standard deviations of the radii between the previous tracking result and of particle $i$; $d_i$ represents the difference between the ratios of black pixels to white pixels in the circle detected by the previous moment tracking result and in particle $i$. In addition, these quantities must be individually divided by the square of their characteristic difference.

**Gaze-point calculation and correction**

After calculating the iris center point coordinates in the image, it is possible to evaluate the correlation between the center point of the iris and the coordinates for the gaze point on the screen. Figure 14 illustrates the result of the subject gazing at the nine points on the screen for gaze-point correction.

The location of the green dot on the screen varies depending on the subject’s gaze direction. The center coordinates of the iris in the image also differ. Numerous methods exist for correction mapping. For example, the method of calculating the gaze-point map by using the first-order polynomial equation is simple, and few resources are required for calculation, but the accuracy is low.25 A fifth-order polynomial equation is used for calculation when using the gaze-point mapping method, thus improving accuracy and computational complexity;26 the most commonly used method is gaze-point mapping through second-order polynomial equations.27 Because eye movements are nonlinear, this study used the second-order polynomial method.

In formula (21), $S_x$ and $S_y$ are the coordinate points on the screen; $x$ and $y$ are the human eye images captured by the camera. Converted through 12 unknown parameters such as $a_1$ to $a_6$, and $b_1$ to $b_6$, these 12 unknown parameters are obtained using the nine-point corrected gaze points. This study used the singular value decomposition28
method. The results of the gaze-point mapping appear on the screen after completion of the calibration procedure. The calibration process only needs to be performed once before the initial use.

**Experiment results**

**Experiment device and environment**

Figure 15 presents an iris center detection result diagram. This study conducted three experiments, namely for evaluating the performance of eye detection, iris center locating, eye tracking, and fixation error calculation. In these experiments, seven subjects were tested with normal, weak, and strong light. Users looked at nine points on the screen and moved their heads to make the light change. This study collected data on precision, recall, and accuracy. In addition, this study conducted experiments on iris center gap calculation and fixation error calculation. Please refer to Table 1 for detailed specifications and a description of relevant computers.

Please refer to Table 2 for detailed specifications of the Acer V276HL screen, which had a resolution of 1920 × 1080 pixels.

This study was conducted in one environment for all seven subjects, who were presented with the same scene. The test items were open eye detection rate and eye-tracking accuracy. The detection rate had four calculation parameters: true positive (TP), which indicates that both the actual situation and detection are positive samples; true negative (TN), which indicates that both the actual situation and detection are negative samples; false positive (FP), which indicates that the actual situation is a negative sample but the detection is a positive sample; false negative (FN), which indicates that the actual situation is a positive sample but is detected as a
negative sample. If a positive sample actually occurs, then the event has a value of 1 and the negative sample is zero. If a positive sample is detected, then the detection is 1 and the negative sample is zero.

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (22)
\]

\[
\text{Precision} = \frac{TP}{TP + FP} \quad (23)
\]

\[
\text{Recall} = \frac{TP}{TP + FN} \quad (24)
\]

In formula (22), the accuracy is the ratio of the correct detection of all samples. In formula (23), the precision is the ratio of positive samples of events in all positive samples detected. In formula (24), the recall is the ratio of the positive samples detected in the positive samples of all events.

In the experiment, this study uses the webcam at 30 fps with a resolution of 320 pixels × 240 pixels. In the course of the experiment, the detection speed has reached 0.015 s and it is real time. If that can use a better camera, you can get better images.

Kao et al., Jen et al., and Dobeš et al. are evaluated under similar scenarios with this study. So this study is compared with the Kao et al., Jen et al., and Dobeš et al. Only provided the performance evaluation results on eye detection, Kao et al. only provided performance evaluation results on locating the iris centers; while Jen et al. provided the performance evaluation results on both eye detection and tracking. Therefore, the method of this study was compared with the methods of Jen et al. and Dobeš et al. for performance evaluation on eye detection results; and compared with the methods of Jen et al. and Kao et al. for tracking results.

As compared with the above-mentioned literatures, the average errors of the eye detection of this study are stable within 9 pixels. The average error of the fixation mapping is within 3° within 50 cm of the screen, and a low-cost and low-speed camera can also be adoptable for eye detection under different lighting sources.

**Performance evaluation on eye detection**

This experiment is comparable to those conducted by Jen et al. and Dobeš et al. Figure 16 presents an experiment conducted in a room with normal lighting. Users looked at nine points on the screen and were able to move their heads suitably.

Table 3 lists the results of open eye detection rate under normal lighting conditions for seven users, which reveal that the average accuracy, precision, and recall rates can exceed 97% under normal lighting conditions.

| Table 1. Computer details. |
|---------------------------------------------------------------|
| **Type** | **Item** | **Specification** |
| Hardware | CPU | I7-4770 3.4 GHz |
|          | RAM | 16 GB |
| Software | OS | Windows 8 |
|          | Developing language | Visual Studio 2015/C++ |
|          | Compiler | Intel C++ 9.1 |

| Table 2. Screen details. |
|--------------------------|
| **Screen pixel** | 1920 pixels × 1080 pixels |
| **Screen size** | 59.5 cm × 33.4 cm |
| **Pixel : cm** | 1 : 0.031 |
| **Image pixel** | 320 pixels × 240 pixels |
| **Distance between screen and user** | 50 cm |
In addition, other rates were higher than those reported by Jen et al.\textsuperscript{12} and Dobeš et al.,\textsuperscript{13} except for precision, which was slightly lower than that reported by Dobeš et al.\textsuperscript{13}

Figure 17 demonstrates the setting of experiments performed under weak lighting conditions. The environment and subjects were the same in this experiment, but all indoor lights were turned off and the only light source was a window. Users looked at nine points on the screen and moved their heads to make the light change.

Table 4 lists the open eye detection rates under weak lighting conditions. Although the average result was unfavorable compared with that reported by Jen et al.\textsuperscript{12} and precision was lower than that reported by Dobeš et al.,\textsuperscript{13} the average accuracy, precision, and recall rates of eye detection results under weak lighting conditions reached 98.55\%, 99.17\%, and 97.75\%, respectively; thus, the experiment demonstrated satisfactory open eye detection rates.

Figure 18 demonstrates the setting of experiments performed under strong lighting conditions. Users looked at nine points on the screen and moved their heads to make the light change.

![Figure 16. Test under normal lighting conditions.](image)

### Table 3. Eye detection rates under normal lighting conditions.

| User   | Frame | TP   | TN | FP | FN | Recall (%) | Precision (%) | Accuracy (%) |
|--------|-------|------|----|----|----|------------|---------------|--------------|
| User1  | 504   | 503  | 0  | 0  | 1  | 99.80      | 100.00        | 99.80        |
| User2  | 618   | 597  | 7  | 14 | 0  | 97.21      | 98.76         | 97.73        |
| User3  | 581   | 557  | 1  | 7  | 16 | 99.83      | 98.63         | 98.90        |
| User4  | 590   | 578  | 3  | 8  | 1  | 98.90      | 98.75         | 98.68        |
| User5  | 646   | 630  | 1  | 8  | 7  | 98.88      | 99.61         | 97.51        |
| User6  | 522   | 509  | 0  | 2  | 11 | 98.73      | 98.60         | 97.36        |
| User7  | 721   | 702  | 0  | 10 | 9  | 98.91      | 98.87         | 97.84        |
| Average |       |      |    |    |    | 98.91      | 98.87         | 97.84        |
| Jen et al.\textsuperscript{12} |       |      |    |    |    | 96.86      | 98.71         | 96.57        |
| Dobeš et al.\textsuperscript{13} |       |      |    |    |    | 94.43      | 98.98         | 93.88        |

TP: true positive; TN: true negative; FP: false positive; FN: false negative.
As listed in Table 5, the open eye detection rate with strong light demonstrated average results superior to those reported by Jen et al. and Dobe et al. The average accuracy, precision, and recall rates exceeded 98%.

**Performance evaluation of locating the iris center**

Figure 19 presents the diagram of the experiment performed for locating the iris center, which is comparable to those performed by Kao et al. and Jen et al. The figure has three points representing different studies. The green circle represents the true center of the iris, the yellow rectangle represents the iris center detected by Kao et al., the pink cross represents the iris center detected by Jen et al., and the blue triangle represents the iris center detected in the present study.

As shown in Table 6, normal, weak, and strong lighting conditions produced different iris detection results. The experimental results revealed that the performance of iris center locating was worse under normal and strong lighting conditions than under weak lighting conditions. Normal lighting demonstrated an average distance of...
Figure 18. Test under strong lighting conditions.

Table 5. Open eye detection rates under strong lighting conditions.

| User  | Total frame | TP   | TN  | FP  | FN  | Recall (%) | Precision (%) | Accuracy (%) |
|-------|-------------|------|-----|-----|-----|------------|---------------|--------------|
| User1 | 938         | 938  | 0   | 0   | 0   | 100.00     | 100.00        | 100.00       |
| User2 | 833         | 792  | 10  | 14  | 17  | 97.90      | 98.26         | 96.28        |
| User3 | 1060        | 1058 | 0   | 0   | 2   | 99.81      | 100.00        | 99.81        |
| User4 | 1201        | 1121 | 2   | 25  | 53  | 95.49      | 97.82         | 95.42        |
| User5 | 1227        | 1189 | 11  | 23  | 4   | 99.66      | 98.10         | 97.80        |
| User6 | 1122        | 1075 | 14  | 30  | 3   | 99.72      | 97.29         | 98.48        |
| User7 | 682         | 661  | 5   | 16  | 0   | 100.00     | 97.64         | 99.27        |
| Average|             |      |     |     |     | 98.94      | 98.44         | 98.15        |
| Jen et al. |         |      |     |     |     | 96.47      | 98.10         | 96.24        |
| Dobeš et al. |     |      |     |     |     | 93.67      | 98.16         | 93.39        |

TP: true positive; TN: true negative; FP: false positive; FN: false negative.

Figure 19. Diagram of locating the iris center.
6.64 pixels, and strong lighting demonstrated an average distance of 6.95 pixels. Weak light produces less reflective light on the iris and more accurate detection of the iris center. In addition, the experimental results indicate that the iris center was more accurately located in the present study than in those conducted by Kao et al.\(^4\) and Jen et al.\(^12\).

**Performance evaluation on eye tracking**

Figure 20 presents the comparative eye-tracking results. This experiment compares to those performed by Kao et al.\(^4\) and Jen et al.\(^12\). The experimental also conducted under normal, weak, and strong lighting conditions. As shown in Table 7, normal, weak, and strong lighting conditions produced different eye tracking error results. The experimental results revealed that the eye-tracking errors were the best under weak lighting conditions among all other lighting conditions. Normal lighting demonstrated an average eye-tracking error distance of 7.15 pixels, while the strong lighting demonstrated an average eye-tracking error distance of 5.16 pixels. Weak light produces the best performance on the eye-tracking results. In addition, the experimental results also indicate that the proposed eye-tracking method can achieve better performance than those conducted by Kao et al.\(^4\) and Jen et al.\(^12\).

**Performance evaluation of fixation error**

Figure 21 presents the fixation mapping diagram. The subject is given 6 s to stare at each point; the data were divided into three sets. Each data set has central coordinates of the iris for a subject that gazed at nine points. The system must randomly select a datum to calculate the gaze-point coordinate conversion parameters. The remaining two data sets are used with calculated parameters to convert the iris center coordinates into gaze-point coordinates. The system must then select one of the remaining two materials to calculate the conversion parameters after the statistics have been obtained before using this calculated parameter to convert the iris center coordinates of the remaining two data sets. This is the loop that must be iterated until an answer is obtained for each data set. As listed in Table 8, each subject produced an average gaze-point mapping error. The error angles of the sixth, seventh, and eighth points were greater than 3°, but the angles of the remaining

| User | Normal lighting (pixels) | Weak lighting (pixels) | Strong lighting (pixels) |
|------|--------------------------|------------------------|--------------------------|
| User1 | 6.63                     | 4.39                   | 5.58                     |
| User2 | 6.37                     | 4.68                   | 7.77                     |
| User3 | 5.53                     | 5.37                   | 7.37                     |
| User4 | 5.69                     | 3.72                   | 8.08                     |
| User5 | 6.68                     | 4.12                   | 7.70                     |
| User6 | 7.12                     | 4.17                   | 4.27                     |
| User7 | 8.45                     | 4.59                   | 7.85                     |
| Average | 6.64                   | 4.43                   | 6.95                     |
| Kao et al.\(^4\) | 85.32                   | 88.74                 | 89.89                   |
| Jen et al.\(^12\) | 11.71                   | 10.24                 | 16.96                   |

Figure 20. Diagram of eye tracking.
points were within 3\(^\circ\). Therefore, the overall accuracy of the mapping was higher than that of mapping performed in other studies.

**Summary of experiment results**

In eye detection rates under normal lighting conditions. The average accuracy, precision, and recall rates can exceed 97\% under normal lighting conditions. In addition, other rates were higher than those reported by Jen et al.\(^{12}\) and Dobeš et al.,\(^{13}\) except for precision, which was slightly lower than that reported by Dobeš et al.\(^{13}\) In the open eye detection rate under weak lighting conditions. Although the average result was unfavorable compared with that reported by Jen et al.\(^{12}\) and precision was lower than that those reported by Dobeš et al.\(^{13}\) In the open

**Table 7. Differences in eye-tracking error.**

|                      | Normal lighting (pixels) | Weak lighting (pixels) | Strong lighting (pixels) |
|----------------------|--------------------------|------------------------|--------------------------|
| This study           | 7.15                     | 4.80                   | 5.16                     |
| Kao et al.\(^{4}\)   | 90.49                    | 103.99                 | 65.45                    |
| Jen et al.\(^{12}\)  | 12.93                    | 14.99                  | 24.38                    |

**Table 8. Average fixation mapping error.**

| Error     | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | Average |
|-----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|---------|
| Distance (pixels) | 54.27 | 57.28 | 70.61 | 84.78 | 83.06 | 98.54 | 93.21 | 110.15 | 87.66 | 82.17   |
| Angle (degree)    | 1.65 | 1.95 | 2.12 | 2.69 | 2.79 | 3.14 | 3.13 | 3.63 | 2.73 | 2.65    |
| X (pixels)        | 39.50 | 36.03 | 37.39 | 34.70 | 52.57 | 63.00 | 36.93 | 35.46 | 46.26 | 42.43   |
| Y (pixels)        | 28.88 | 36.42 | 54.20 | 71.26 | 54.64 | 67.00 | 79.23 | 100.50 | 66.98 | 62.12   |

**Figure 21. Fixation mapping diagram.**
eye detection rate under strong lighting conditions. The open eye detection rate with strong lights demonstrated the average results of this study are superior to those reported by Jen et al.\textsuperscript{12} and Dobes et al.\textsuperscript{13} As for the differences in iris center locating performance, the performance of iris center locating was worse under strong lighting conditions than under weak lighting conditions. The iris centers were more accurately located in the present study than in those conducted by Kao et al.\textsuperscript{4} and Jen et al.\textsuperscript{12} In part of differences in eye-tracking error, the experimental results revealed that the eye-tracking errors were the best under weak lighting conditions among all other lighting conditions that the proposed eye-tracking method can achieve better performance than those conducted by Kao et al.\textsuperscript{4} and Jen et al.\textsuperscript{12} In part of the average fixation mapping error, the overall accuracy of the mapping results of this study was higher than that of mapping performed in other studies.

As a result, the proposed method can achieve an eye tracker in a low cost and simple way. So that the user can conduct interactive control signals by moving the eye movements. Simple instructions are given to achieve HMI results.

**Conclusion**

This study develops an eye-tracker system for HMI and interactive machine. The eye-tracker system proposed by the research institute is used with natural light. The system can be safely used for a long period of time without placing extra burden on the eyes. It is cheaper than most eye trackers on the market because it uses a general webcam. The experimental results prove that the overall open eye detection precision and accuracy rate exceeded 95\% and that the error in detecting the iris center error is also small. The fixation error calculation revealed that the angles of the remaining points were within 3° of each other. This study compared with the methods of Jen et al.\textsuperscript{12} and Dobes et al.\textsuperscript{13} which are performance evaluation on eye detection. The method of this study compared with the methods of Jen et al.\textsuperscript{12} and Kao et al.\textsuperscript{4} which are performance evaluation of locating the iris center. Therefore, the overall mapping accuracy is high. This study designed an eye tracker that only requires a webcam and computer to track the line of fixation and map the gaze point of the user. In addition, all of the hardware components used were inexpensive and easy to obtain. The overall system cost is lower than that of other products on the market.

**Acknowledgement**

This manuscript was edited by Wallace Academic Editing.

**Declaration of conflicting interests**

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

**Funding**

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by Ministry of Science and Technology of Taiwan under the grant number MOST 107–2218E-027–018, and MOST 107–2622-E-027–022-CC2.

**ORCID iD**

Yen-Lin Chen \( \text{https://orcid.org/0000-0001-7717-9393} \)

**References**

1. Jira JC, Wingrove TC and Channey RD. Finger pointing, gesture based human-machine interface for vehicles. U.S. Patent No. 8,760,432, 24 June 2014.
2. Research—Eye tracker and brainwave operation sprout in the market of the frozen people, DIGITIME, 2017, http://www.digitimes.com.tw/tech/rpt/rpt_show.asp?cnlid=3&pro=y&priname=%a5%ab%b3%f5&cat=pec&v=20140906-355 (accessed 23 August 2017).
3. What role does eye tracking technology play in VR? TechNews, 2017, https://technews.tw/2016/01/14/eye-head-coordination-for-visual-cognitive-processing/ (accessed 23 August 2017).
4. Kao WC, Lee CY, Lin CY, et al. Gaze tracking with particle swarm optimization. 2015 International Symposium on Consumer Electronics (ISCE), Madrid, 2015, pp. 1–2.
5. Świrski L and Dodgson NA. A fully-automatic, temporal approach to single camera, glint-free 3D eye model fitting. In Proceedings of the 17th European Conference on Eye Movements (ECEM 2013). http://www.cl.cam.ac.uk/research/rainbow/projects/eyemodelfit/

6. Huey EB. Preliminary experiments in the physiology and psychology of reading. *Am J Psychol* 1897; 9: 575–586.

7. Babcock JS, Marianne L and Pelz JB. How people look at pictures before, during, and after scene capture: Buswell revisited. In SPIE Proceedings Vol. 4662, *Human vision and electronic imaging VII*, 2002, pp. 34–47.

8. Sirohey S, Rosenfeld A and Duric Z. A method of detecting and tracking irises and eyelids in video. *Patt Recognit* 2002; 35: 1389–1401.

9. Yang J, Stiefelhagen R, Meier U, et al. Real-time face and facial feature tracking and applications. In AVSP’98 *International conference on auditory-visual speech processing*, Terrigal, Australia, 1998, pp. 79–84.

10. Sigut J and Sidha S. Iris center corneal reflection method for gaze tracking using visible light. *IEEE Trans Biomed Eng* 2011; 58: 411–419.

11. Lee H, Iqbal N, Chang W, et al. A calibration method for eye-gaze estimation systems based on 3D geometrical optics. *IEEE Sens J* 2013; 13: 3219–3225.

12. Jen CL, Chen YL, Lin YJ, et al. Vision based wearable eye-gaze tracking system. 2016 *IEEE international conference on consumer electronics (ICCE)*, Las Vegas, NV, 2016, pp. 202–203.

13. Dobes M, Martinek J, Skoupila D, et al. Human eye localization using the modified Hough transform. *Optik Int J Light Electr Optics* 2006; 117: 468–473.

14. Wu BF, Chen YL and Chiu CC. A discriminant analysis based recursive automatic thresholding approach for image segmentation. *IEICE Trans Informat Syst* 2005; 88: 1716–1723.

15. Chen YL, Liang WY, Chiang CY, et al. Vision-based finger detection, tracking, and event identification techniques for multi-touch sensing and display systems. *Sensors* 2011; 11: 6868–6892.

16. Chen YL, Chiang HH, Chiang CY, et al. A vision-based driver nighttime assistance and surveillance system based on intelligent image sensing techniques and a heterogamous dual-core embedded system architecture. *Sensors* 2012; 12: 2373–2399.

17. Kim YT. Contrast enhancement using brightness preserving bi-histogram equalization. *IEEE Trans Cons Electr* 1997; 43: 1–8.

18. Leavers VF. *Shape detection in computer vision using the Hough transform*. Berlin: Springer-Verlag, 1992.

19. Gander W, Golub GH and Strebel R. Least-squares fitting of circles and ellipses. *BIT Numer Math* 1994; 34: 558–578.

20. Yun X and Bachmann ER. Design, implementation, and experimental results of a quaternion-based Kalman filter for human body motion tracking. *IEEE Trans Robot* 2006; 22: 1216–1227.

21. Xu F, Liu X and Kikuo F. Pedestrian detection and tracking with night vision. *IEEE Trans Intell Transp Syst* 2005; 6: 63–71.

22. Wenga S, Kuo C and Tub S. Video object tracking using adaptive Kalman filter. *ELSEVIER Trans Vis Commun Image* 2006; 17: 1190–1208.

23. Wren CR, Azarbayejani A, Darrell T, et al. Pfrender: real-time tracking of the human body. *IEEE Trans Pattern Anal Machine Intell* 1997; 19: 780–785.

24. Li Z, Liu and H and Xu C. Real-time human tracking based on switching linear dynamic system combined with adaptive Meanshift tracker. *IEEE international conference on image processing*, Brussels, 2011, pp. 2329–2332.

25. White JKP, Hutchinson TE and Carley JM. Spatially dynamic calibration of an eye-tracking system. *IEEE Trans Syst Man Cybern* 1993; 23: 1162–1168.

26. Cherif ZR, Naït-Ali A, Motsch JF, et al. An adaptive calibration of an infrared light device used for gaze tracking. *IMTC* 2002. *Proceedings of the 19th IEEE instrumentation and measurement technology conference (IEEE Cat. No.00CH37276)*, Anchorage, AK, USA, 2002, pp. 1029–1033, vol. 2.

27. Ramanauskas N. Calibration of video-oculographical eye-tracking system. *Elektronika ir Elektrotechnika* 2006; 8: 65–68.

28. Gene GH and Reinsch C. Singular value decomposition and least squares solutions. *Numer Math* 1970; 14: 403–420.