Learning gaze biases with head motion for head pose-free gaze estimation

Feng Lu[a,] , Takahiro Okabe[b], Yusuke Sugano[a], Yoichi Sato[a]

[a] The University of Tokyo, Japan
[b] Kyushu Institute of Technology, Japan

abstract

When estimating human gaze directions from captured eye appearances, most existing methods assume a fixed head pose because head motion changes eye appearance greatly and makes the estimation inaccurate. To handle this difficult problem, in this paper, we propose a novel method that performs accurate gaze estimation without restricting the user's head motion. The key idea is to decompose the original free-head motion problem into subproblems, including an initial fixed head pose problem and subsequent compensations to correct the initial estimation biases. For the initial estimation, automatic image rectification and joint alignment with gaze estimation are introduced. Then compensations are done by either learning-based regression or geometric-based calculation. The merit of using such a compensation strategy is that the training requirement to allow head motion is not significantly increased; only capturing a 5-s video clip is required. Experiments are conducted, and the results show that our method achieves an average accuracy of around 3° by using only a single camera.

Keywords:
Gaze estimation
Free head motion
Head pose compensation
Appearance-based approach

1. Introduction

Human gaze plays an essential role in conveying human visual attention, desire, feeling, intention, and so on [1]. Therefore, research on human gaze estimation has been an important topic since decades ago [2] and has attracted much attention in recent years. There are already systems in the market that have been used in many applications, such as human..computer interfaces, cognitive study, market research, and driver training. However, these existing systems usually require expensive devices, which stop them from being widely used by ordinary users in their everyday lives.

With the fast development of computer vision technology, it seems possible that human gaze direction can be estimated by only using captured eye images without requiring too much additional hardware. This will greatly enhance the applicability of gaze estimation technology. According to recent surveys [3,4], existing computer vision-based methods can be roughly divided into two categories: model-based methods and appearance-based methods. Methods in the first category assume 3D or 2D eyeball models to estimate gaze directions, while they still need some additional devices such as infrared lights and cameras. Therefore, they are more suitable for use in a controlled environment such as in the laboratory. Appearance-based methods have the advantage of using only a single common camera, and thus, they attract increasing attention.

In this paper, we focus on the appearance-based methods and solve the major difficulty in allowing free head motion. This problem is difficult because head motion changes eye appearance greatly, and therefore, gaze estimation with changed eye appearances will be inaccurate. A straightforward way to solve this problem is to store all the eye appearances for every individual head pose in the system. However, because head motion has 6 degrees of freedom, it becomes impractical to directly deal with all possible eye appearances due to head motion.

The key idea of this paper is to decompose the originally difficult problem into simple subproblems and solve them efficiently. In particular, instead of considering gaze estimation for arbitrary head poses, we first apply gaze estimation by assuming a fixed head pose, and then compensate for the estimation biases caused by the head pose difference. The compensation comprises two stages, namely, eye appearance distortion compensation and geometric compensation. We propose methods to complete these compensations and show that the gaze estimation can be done accurately for arbitrary head poses.

The primary contributions of this work are:

(1) A gaze estimation approach that decomposes the difficult free head motion problem into much easier subproblems is proposed.
(2) Subproblems are solved with our proposed compensation methods, which correct gaze estimation biases due to both eye appearance distortion and geometric factors.
(3) The training cost for compensating for head motion is low. It only requires capturing a short video clip of user free head motion for a duration of 5 s.

(4) Eye images captured under different head poses are robustly aligned via image rectification and iterative optimization.

Overall, compared with existing appearance-based methods, our method does not use any other devices, while it allows for head motion by only requiring an additional calibration step to capture a short video clip. If compared with existing model-based methods that also allow for head motion, our method only uses a common camera with known location and intrinsic parameters, while most of the others use multiple cameras/lights and therefore require more complicated camera/geometric calibrations.

2. Related works

Computer vision-based human gaze estimation has been undergoing rapid development. According to recent surveys [3,4], existing computer vision-based methods can be roughly divided into two categories: model-based methods and appearance-based methods. Methods belonging to the former category extract and use very small eye features in the captured eye images, such as the reflection points in the corneal surface [5..9], pupil center [10,11], and iris contour [12]. These features are used to fit a 2D or 3D eyeball model to geometrically calculate the gaze direction regardless of the head pose. For instance, Beymer and Flickner [13] proposed generating and detecting corneal reflections via zoom-in cameras and infrared LEDs equipped on stereo pan-tilt units. Additionally, two additional wide range stereo cameras are used for eye position tracking. Brolly and Mulligan [14] proposed a similar method for accurately extracting and using corneal reflection points, as well as did Nagamatsu et al. [15] and Zhu and Ji [7]. To reduce number of cameras, Villanueva and Cabeza [8] suggested using more infrared LEDs to achieve accurate geometric calculation. Yoo and Chung [5] proposed a novel method based on the cross-ratio that computes the 3-D relationship of the eyeball, cameras, and screen. Kang et al. [16] further improved the cross-ratio method by taking into account the individual eyeball parameter differences.

Although model-based methods show advantages in handling head motion and so on, they also have limitations. First, it is not easy to extract small eye features in the eye. Therefore, high resolution and even infrared imaging is always needed. Second, many of the previous works further require special cameras and mechanical units [13,14,17] to allow for head motion, which are not applicable for ordinary users in their everyday lives.

On the contrary, appearance-based methods have the advantage of requiring only a single camera that works under natural illumination. They only use images with common or even low resolutions. They typically regard a whole eye image as a high-dimensional input vector and learn the mapping between these vectors and the gaze positions. Early methods such as the neural networks proposed by Baluja and Pomerleau [18] and Xu et al. [19] need thousands of labeled training samples to train the mapping.

Tan et al. [20] proposed constructing an eye appearance manifold by using 252 collected training samples and using the local structure of the manifold gaze for interpolation. To reduce the number of labeled training samples, Williams et al. [21] introduced a semi-supervised Gaussian process regression to use both labeled and unlabeled training samples. Sugano et al. [22] investigated image saliency and proposed a user-unaware calibration that is performed while a user is watching a video clip. This idea was also used for a model-based method [17]. Lu et al. [23] introduced a novel regression technique to use sparse training samples and achieve high accuracy gaze estimation.

However, allowing for head motion is more difficult for appearance-based methods. Simply performing gaze estimation under head motion results in significantly large errors [24,25]. Nguyen et al. [26] proposed collecting training images for different head poses, which results in a lengthy calibration. Sugano et al. [27] proposed re-collecting training sample incrementally while the user was naturally using a computer. Lu et al. [28] introduced a synthesis framework to produce training eye images under unseen head poses. This method estimates optical flows and synthesizes a fair amount of training images, and therefore, its computational cost may be too high for mobile devices. In summary, because of the limitations in calibration and computational cost, the problem of head motion remains challenging for existing appearance-based methods.

Table 1
Definitions of notations used in this paper.

Notation	Description
$e \in \mathbb{R}^m$	Eye appearance vector comprising m pixels
$r = [r^x, r^y, r^z]^T \in \mathbb{R}^3$	3D head rotation vector
$t = [t^x, t^y, t^z]^T \in \mathbb{R}^3$	3D head translation vector
$g = [g^x, g^y, g^z]^T \in \mathbb{R}^3$	3D gaze direction vector
$\{\hat{e}, \hat{r}, \hat{t}, \hat{g}\}$	Data for a test sample
$T^g = \{e_i \| i = 1, \cdots, n\}$	Training dataset of eye appearance
$T^g = \{r_i \| i = 1, \cdots, n\}$	Training dataset of head rotation
$T^g = \{t_i \| i = 1, \cdots, n\}$	Training dataset of head translation
$T^g = \{g_i \| i = 1, \cdots, n\}$	Training dataset of gaze direction
$T = \{T^e, T^r, T^t, T^g\}$	Complete training dataset
r_0, t_0	Rotation and translation of a fixed head pose
$T_0 = \{T_0^e, T_0^g\}$	Training dataset corresponding to r_0 and t_0

3. Overview of the approach

3.1. Problem with head motion

Some important notations are first given in Table 1. Using these notations, the gaze estimation problem can be represented as:

$$\hat{g} = E(\hat{e}, \hat{r}, \hat{t} \,|\, T), \tag{1}$$

where $E(\cdot)$ is a function for estimating the gaze direction $\hat{g}$[1] from the eye appearance $\hat{e}$ by using the training dataset $T = \{T^e, T^r, T^t, T^g\}$. Throughout this paper, the letters e, r, t, and g are always used to indicate eye image vector, head rotation, head translation, and gaze direction, respectively. Note that the problem defined in Eq. (1) does not assume a fixed head pose; therefore, the head pose parameters $\hat{r}$ and $\hat{t}$, which are input head rotation and head translation vectors in a 3D coordinate system, are also required as inputs of the problem. Also note that strictly speaking, 3D gaze is defined by its 3D direction and 3D origin. In this paper, we focus on its 3D direction because the 3D origin is just the eye position that can be directly tracked via a head/eye tracker.

In this paper, we solve the problem defined by Eq. (1). As shown in Fig. 1, our goal is to estimate the gaze direction $\hat{g}$ under the world coordinate system (WCS), while the head coordinate system (HCS) determined by head pose $(\hat{r}, \hat{t})$ can be arbitrary due to head motion.

3.2. Decomposition of the problem

The free head motion problem defined in Eq. (1) is difficult because it requires the training dataset T, which corresponds to 6-degree-of-freedom head motion to sufficiently train the mapping function $E(\cdot)$. Therefore, as reviewed in Section 2, most existing appearance-based methods assume a fixed head pose. In this paper, we propose decomposing this difficult problem into much easier subproblems that can be solved by using less training data.

We first consider the problem by assuming eye images captured under a fixed head pose (r_0, t_0). Then the problem can be simplified as

$$\hat{g} = E_{r_0,t_0}(\hat{e} \,|\, T_0), \tag{2}$$

[1] In practice, gaze estimation methods usually estimate the 2D gaze position on the screen instead of the 3D gaze direction vector for convenience. Such gaze positions and gaze directions can be converted into each other after geometric calibration.

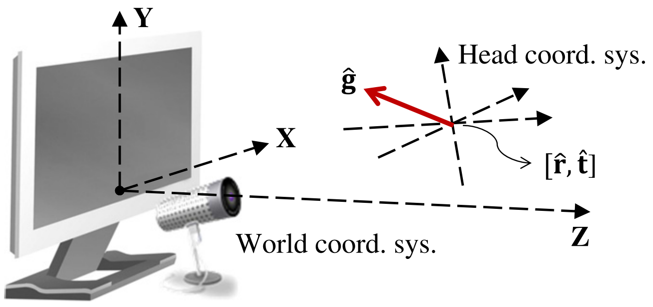

Fig. 1. Gaze estimation with head motion. Gaze direction $\hat{g}$ is estimated under world coordinate system with arbitrary head rotation $\hat{r}$ and translation $\hat{t}$.

where $T_0 = \{T_0^e, T_0^g\}$ is the training dataset collected for the fixed head pose (r_0, t_0), and function $E_{r_0,t_0}(\cdot)$ performs gaze estimation only for such a fixed head pose. This subproblem can be easily solved by using previous methods to give an initial solution.

However, the above gaze estimation may contain significant error if the head moves. Fig. 2 gives an example with temporary notations α, β, etc. As shown in Fig. 2(a), the estimated gaze direction under a fixed head pose, r_0, t_0, is α. Then, if the head pose moves to $(\hat{r}, \hat{t})$ as in Fig. 2(b), the estimated gaze direction α will differ from the true gaze direction by a bias, β. Therefore, we need to compensate for α by using β to obtain the correct gaze direction. This leads to the idea of gaze compensation.

Furthermore, compensation for gaze bias can be further decomposed into two stages, as shown in Fig. 2(c). First, α differs from α because head motion changes the camera's viewing direction and thus distorts the eye image (see the small eye image in Fig. 2). This causes an estimation bias, β^D. Second, the gaze direction rotates with head rotation geometrically, and this causes another bias, β^G. Note that β^G has a precise closed-form solution that can be geometrically computed, while β^D does not. For this reason, in this paper, we treat the two compensations separately, and therefore, uncertainty mainly goes to β^D; thus, the total error is limited. Another reason to do so is that computing β^G only requires head rotation angles as inputs, while β^D depends on the camera's viewing direction that is related to all head pose parameters. Therefore, it is better to treat these two compensation problems separately.

By considering the above compensations, the problem in Eq. (1) can be decomposed into

$$\hat{g} \simeq E_{r_0,t_0}(\hat{e}|T_0) \otimes C^D_{r_0,t_0}(\hat{r}, \hat{t}|T) \otimes C^G_{r_0}(\hat{r}), \tag{3}$$

where $E_{r_0,t_0}(\hat{e}|T_0)$ indicates an initial gaze direction by assuming a fixed head pose, (r_0, t_0). Operator •⊗Žapplies manipulations on the gaze direction via a series of rotations, and $C^D_{r_0,t_0}(\hat{r}, \hat{t}|T)$ and $C^G_{r_0}(\hat{r})$ denote the specific compensations for eye appearance distortion and geometric bias.

Finally, Fig. 3 summarizes our decomposition scheme for handling the problem and briefly shows the keys of our compensation techniques. Method to obtain the initial solution is described in Section 4 and then details of our compensation techniques are described in Section 5. In particular, eye appearance distortion compensation in Section 5.1 uses a special set of training samples obtained via a novel calibration method. The calibration obtains sufficient training samples in a clever way by only capturing a very short video clip of the user. This video is the only additional training cost of our method, compared with conventional appearance-based methods that do not allow for head motion.

3.3. Implementation and procedures

A head-pose free gaze estimation system can be implemented on the basis of the proposed compensation approach. Only a single camera is used to capture the user's eye appearances without requiring any additional devices. To obtain head poses, a commercial head pose tracker [29] is used by our system. It uses the same single camera with known intrinsic/extrinsic parameters, and technical details on it can be found in [29]. Our system works in the following steps.

3.3.1. Obtaining training data

Training dataset $T = \{T^e, T^r, T^t, T^g\}$ is obtained via calibration. In particular, there are two stages of calibration to obtain sufficient training samples including: a fixed head pose calibration and another one done by capturing a short video clip. In general, the user is asked to sit in front of the screen and gaze at specific screen positions, i.e., calibration points. A single camera is placed below the screen. The i-th training sample obtained by the system comprises I_i, g_i, r_i, and t_i, where I_i is the captured face image from which we extract the eye feature e_i.

In the first calibration stage, the user is asked to choose and keep a fixed head pose, (r_0, t_0), and gaze at different calibration points shown on the screen. When he/she clicks the mouse button, his/her face image will be captured. The second calibration stage is novel. The user is asked to gaze at a fixed calibration point on the screen and make head motion to let the camera record a short video clip. Because the gaze position is fixed and the head motion is relatively free, this stage only lasts for several seconds to obtain enough training samples. Details and usages of the training samples from the video are explained later in Section 5.1. Both the two calibration stages should be done for each individual user.

3.3.2. Obtaining test data

During testing, the user can make free head motion and use the mouse to choose any gaze positions on the screen. When he/she clicks a mouse button, the current gaze point position is stored, and a user face image is captured. At the same time, the head pose tracker returns

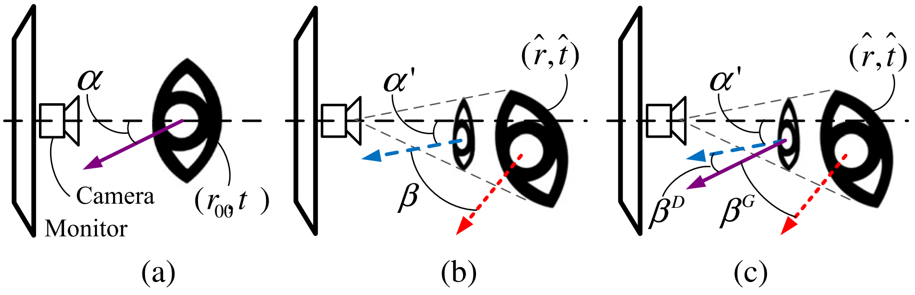

Fig. 2. 2D illustration of problem decomposition. (a) Assuming a fixed head pose (r_0, t_0), gaze direction α can be estimated by Eq. (2). (b) If head pose moves to $\hat{r}, \hat{t}$, captured eye image is distorted; thus, Eq. (2) estimates gaze direction (blue dashed line) as α. It differs from real gaze direction (red dashed line) by β. (c) Difference β can be decomposed into two parts: one due to eye image distortion ($\beta^D = \alpha - \alpha$) and one due to geometric head motion (β^G = head rotation).

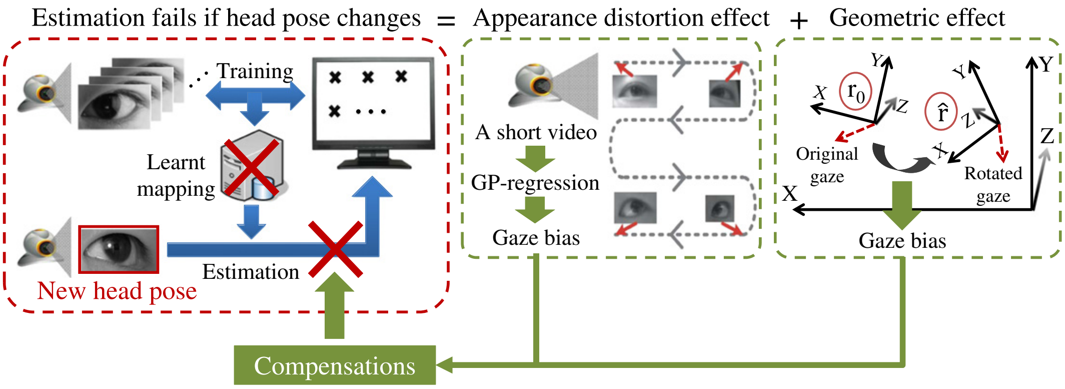

Fig. 3. Problem with head motion and keys of our compensations for handling the problem.

the head pose parameters. In this manner, $\hat{e},\hat{g},\hat{r}$, and $\hat{t}$ are obtained for a test sample.

3.3.3. Gaze estimation

For any test sample, gaze estimation performed by using $\{\hat{e},\hat{r},\hat{t}\}$ and the training dataset T is formulated by Eq. (1) and solved with our proposed techniques. In particular, this problem is first converted into its decomposed form (Eq. (3)) and then solved by techniques introduced in the following sections. Additionally, the gaze direction $\hat{g}$ in the test sample serves as the ground truth for assessing the estimation accuracy.

4. Gaze initialization

Our gaze estimation approach includes two steps: initial gaze estimation and gaze bias compensation, as formulated in Eq. (3) . In this section, we describe how to obtain an initial gaze estimation result by assuming a fixed head pose. This is done by first rectifying the captured images in accordance with the head pose and then performing eye region alignment and fixed-head pose gaze estimation simultaneously.

4.1. Image rectification

Image rectification is unnecessary for existing fixed-head pose gaze estimation methods because they can capture almost identical images with only differences in eyeball orientation. However, when dealing with head motion, the captured user appearances will change drastically, as shown in Fig. 6 (left). This causes great difficulties in appearance-based gaze estimation because the cropped eye regions (shown in the small rectangles) show arbitrary poses. To solve this problem, we rectify the captured images in accordance with the head pose.

The key to rectifying an image is to •hypothetically Žrotate the camera so that the camera coordinate system becomes parallel to the head coordinate system. In this manner, the camera can be regarded as moving in accordance with head motion, and therefore, the captured images look consistent. Examples of the rectified images are shown in Fig. 6 (right), where the cropped eye regions (shown in the small rectangles) have similar poses and can be easily used in gaze estimation.

For the above rectification, our method needs the intrinsic parameters of the camera. These parameters can be estimated via calibration by using a calibration board. This procedure is quick, and it only needs to be done once. The extrinsic parameters of the above hypothetic moving camera do not need to be known for rectification, as shown later.

Rectification is done as follows. After denoting the current head rotation as $r=[r^x,r^y,r^z]^T$, we then need to rotate the camera around the X, Y, and Z axes in turn with the angles $\omega_x=-r^x$, $\omega_y=\tan^{-1}(\tan r^y \cdot \cos r^x)$, and $\omega_z=-r^z$. As a result, the entire rotation matrix can be written as:

$$\boldsymbol{\Omega}_r=\begin{pmatrix}\cos\omega_z & -\sin\omega_z & 0\\ \sin\omega_z & \cos\omega_z & 0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}\cos\omega_y & 0 & \sin\omega_y\\ 0 & 1 & 0\\ -\sin\omega_y & 0 & \cos\omega_y\end{pmatrix}\times\begin{pmatrix}1 & 0 & 0\\ 0 & \cos\omega_x & -\sin\omega_x\\ 0 & \sin\omega_x & \cos\omega_x\end{pmatrix}. \tag{4}$$

Then, we rectify the image by rotating the camera using the rotation matrix $\boldsymbol{\Omega}_r$. If assuming a pin-hole camera model and

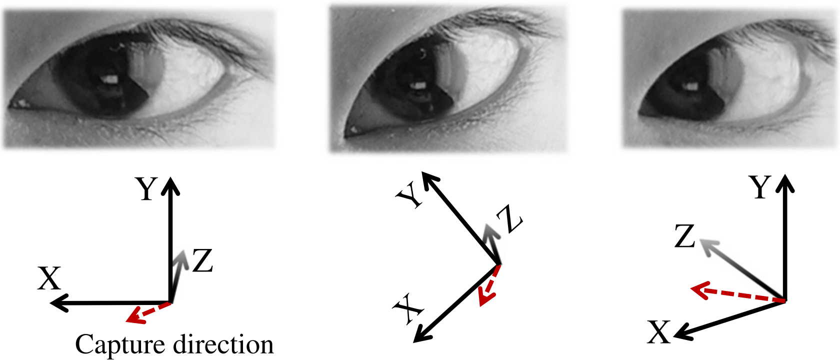

Fig. 4. Eye images captured under different head poses. Head coordinate systems are drawn below each image and viewing direction vectors (dashed arrow) pointing to camera are shown. Note similar appearances of first two images and their similar camera viewing direction vectors under HCS.

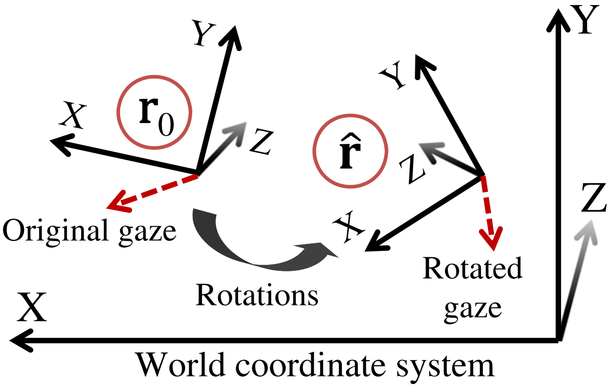

Fig. 5. Rotating gaze vector with head rotation from r_0 to $\hat{r}$.

homogeneous coordinates, any 3D point P is filmed at a 2D image point, p, following

$$\|z\|\begin{pmatrix}p\\1\end{pmatrix} = \boldsymbol{K}[\boldsymbol{RT}]\begin{pmatrix}P\\1\end{pmatrix}, \tag{5}$$

where $\boldsymbol{K} \in \mathbb{R}^{3\times3}$ is the intrinsic matrix of the camera, and $[\boldsymbol{R}|\boldsymbol{T}] \in \mathbb{R}^{3\times4}$ is the extrinsic matrix. Now, we add the above rotation $\boldsymbol{\Omega}_r$ to the camera, and then, the new 2D pixel position $p^{\bullet}$ in the rectified image becomes

$$\begin{aligned}\|z^{\bullet}\|\begin{pmatrix}p^{\bullet}\\1\end{pmatrix} &= \boldsymbol{K}(\boldsymbol{\Omega}_r[\boldsymbol{RT}])\begin{pmatrix}\boldsymbol{P}\\1\end{pmatrix}\\ &= \boldsymbol{K}\boldsymbol{\Omega}_r\boldsymbol{K}^{-1}\left(\boldsymbol{K}[\boldsymbol{RT}]\begin{pmatrix}\boldsymbol{P}\\1\end{pmatrix}\right) = \|z\|\boldsymbol{K}\boldsymbol{\Omega}_r\boldsymbol{K}^{-1}\begin{pmatrix}\boldsymbol{p}\\1\end{pmatrix},\end{aligned} \tag{6}$$

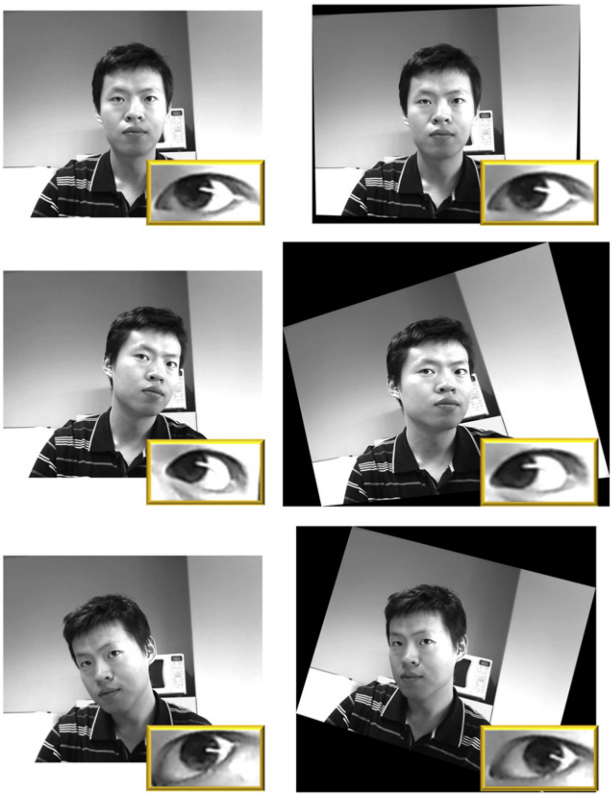

Fig. 6. Eye image rectification. By using projective transformation, captured images (left) are rectified (right) so that eye regions (shown in small rectangles) can be better aligned with each other.

where z and $z^{\bullet}$ are all scalers, and the camera extrinsic parameters $[\boldsymbol{RT}]$ are eliminated.

Note that in Eq. (6), everything is known to transform the original pixel position p into the rectified pixel position $p^{\bullet}$, except for z and $z^{\bullet}$, which are very close to each other and may only scale the image size slightly. By transforming every pixel in the image, we obtain the rectified images shown in Fig. 6 (right). The above rectification is applied to all captured images during both the calibration and estimation procedures. In the next section, we describe how to align eye regions in the rectified images and also handle the small scaling differences, and then we use these eye images to do the initial gaze estimation.

4.2. Initial gaze estimation with eye image alignment

In this section, we describe how to apply the initial gaze estimation by assuming a fixed head pose, as defined by the problem in Eq. (2).

The training data $T_0 = \{T_0^e, T_0^g\}$ obtained under a fixed head pose, (r_0, t_0), are used. This T_0 comprises n_0 training samples, each of which contains a vector, $e_i \in T_0^e$, obtained from the training eye image, I_i and a gaze direction vector, $g_i \in T_0^g$, where $i = 1 \cdots n_0$. For a test sample, we similarly have an eye image vector, $\hat{e}$, and seek its corresponding unknown gaze direction, $\hat{g}$. Note that the head pose parameters $\hat{r}$ and $\hat{t}$ are not considered here since we assume a fixed head pose.

The problem now is about seeking a mapping, $e_i \mapsto g_i$, from the high dimensional space to the 3D gaze direction space. Following the previous fixed-head pose methods [20,23], we solve this problem by using local linear mapping:

$$\begin{aligned}&E_{r_0,t_0}\left(\hat{e}|T_0^e, T_0^g\right) : \hat{g} = \sum_{i=1}^{n_0} w_i \cdot g_i\\ &\quad \text{subject to } \{w_i\} = \arg\min \left\|\hat{e} - \sum_{i=1}^{n_0} w_i \cdot e_i\right\|_F^2,\end{aligned} \tag{7}$$

where the weight $w_i \neq 0$ only when e_i is one of the four closest vectors to $\hat{e}$ in terms of Euclidean distance. Eq. (7) solves the first part, namely, the initial estimation in Eq. (3).

Note that the above method faces a problem in our scenario. Previous fixed-head pose methods capture images under a fixed head pose, and therefore, every eye image can be cropped from the same position to produce an eye image vector, $\hat{e}$. However, in our case, head motion moves eye regions arbitrarily in the captured image, as shown in Fig. 6 (right). We need to align the eye region before extracting the eye image vector $\hat{e}$. Here, we introduce a simultaneous alignment and estimation technique.

One good thing is that the image rectification in Section 4.1 simplifies the alignment problem so that only translation and slight scaling are needed for alignment. In addition, the head pose tracker [29] returns rough eye positions that can be used to initialize the eye image alignment. Therefore, we only need to refine the alignment with high precision. Let $\hat{I}$ denote the captured image, and let $\hat{J}$ be the aligned eye region from $\hat{I}$. Also let $\boldsymbol{\tau}$ and s be the translation and scaling parameters. We use the method in Algorithm 1 to find the test eye region $\hat{J}$ by aligning the reconstructed training image I to the optimal region in $\hat{I}$. The reconstructed training image I is obtained from gaze estimation. In particular, as shown in Eq. (7), we compute a linear combination with weights $\{w_i\}$, and therefore, I is reconstructed from training images by using the same weights $\{w_i\}$.

Note that our alignment method in Algorithm 1 is based on the Lucas..Kanade method [30]. Although their method may fail for extreme movements, it works robustly in our case. The reason is that our head pose tracker already initializes the eye region, and thus, we only need to refine the alignment with a limited number of pixels. Therefore, our alignment can be done robustly, as shown in Fig. 7. Also note that, strictly speaking, alignment for eye images from different head poses cannot be perfect because these images are distorted by 3D head motion. However, our proposed image rectification in Section 4.1 partially handles

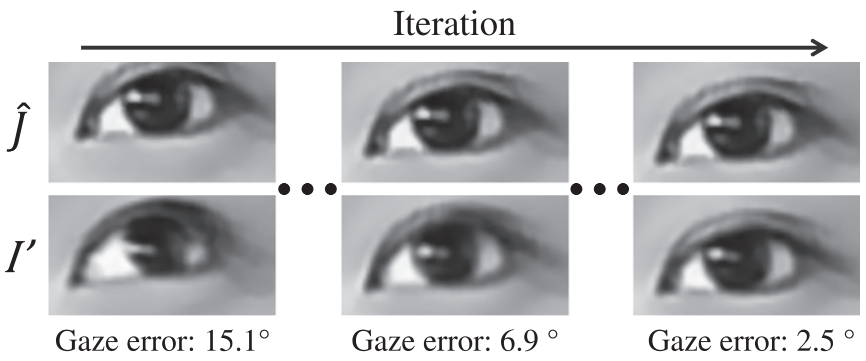

Fig. 7. Simultaneous eye region alignment and gaze estimation. Aligned eye region $\hat{J}$ and reconstructed training eye image I become similar via iterations, and at same time, gaze estimation accuracy is improved.

such distortion, and our proposed appearance distortion compensation in Section 5.1 handles the remaining effect.

Overall, the proposed iterative method refines both gaze estimation and eye image alignment simultaneously and eventually obtains the best solution to Eq. (7) as the initial fixed-head pose gaze estimation result. An example of this procedure is shown in Fig. 7.

Algorithm 1. Simultaneous alignment and estimation

```
Initialize $\boldsymbol{\tau}$ and $s$
while $\boldsymbol{\tau}$ and $s$ have not converged do
    • Crop eye region $\hat{J}$ from $\hat{I}$ by using $\boldsymbol{\tau}$ and $s$
    • Extract $\hat{e}$ from $\hat{J}$ via raster scanning
    • Solve weights $\{w_i\}$ via gaze estimation in Eq. (7)
    • Compute reconstructed image $I = \sum_{i=1}^{n_0} w_i \cdot I_i$
    • Adopt the Lucas‥Kanade method [30] to update $\boldsymbol{\tau}$ and $s$ by
      aligning $I$ to the optimal region in $\hat{I}$
end while
```

5. Gaze compensation

In this section we describe how to compensate for gaze estimation biases existing in the initial gaze estimation (Section 4) due to head motion.

5.1. Compensation 1: eye appearance distortion

In this section we discuss how eye image distortion affects gaze estimation results. Before all, it is necessary to understand how eye images are distorted due to head motion. Fig. 4 shows examples of eye images captured under different head poses. Note that the first two eye images have very similar appearances, although one of them is rotated inside the image plane (this rotation can be rectified as shown in Section 4.1). However, the third eye image's shape is obviously distorted due to the different viewing directions of its camera, and this distortion cannot be rectified by 2D image transformation. This observation leads to the fact that the camera's viewing direction determines the eye appearance distortion.

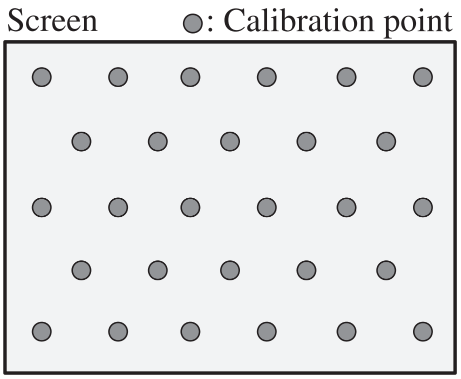

Fig. 8. Calibration points on screen for training with a fixed head pose.

To describe the camera's viewing directions, we represent them under the head coordinate system (HCS). As shown in Fig. 4, these directions are shown as vectors under HCS that point to the camera. The vectors in the first two cases are identical under HCS, indicating that the corresponding eye images have similar 2D appearances, while in the third case, the vector is different, and the eye image is obviously distorted.

We further use the notation $v^c \in \mathbb{R}^3$ to indicate the camera's viewing direction vector under HCS for any head pose. In particular, let v_0^c be a constant camera's viewing direction vector for the fixed head pose r_0, t_0. Then for any head pose, we can compute the difference $\Delta v^c = v^c - v_0^c$. We state that Δv^c is closely related to the gaze estimation bias due to eye image distortion. To represent such a •gaze estimation bias Ž, let $\Delta\boldsymbol{\phi} = [\Delta\phi^x, \Delta\phi^y]$ be rotations around the X and Y axes under WCS as shown in Fig. 1. The physical meaning of $\Delta\boldsymbol{\phi} = [\Delta\phi^x, \Delta\phi^y]$ is that a biased gaze direction can be rotated by $\Delta\phi^x$ and $\Delta\phi^y$ around the X and Y axes to become the correct gaze direction. Therefore, $\Delta\boldsymbol{\phi}$ represents the gaze bias. Then, the problem is about finding the relationship between the difference Δv^c of the camera's viewing direction vectors and the gaze estimation bias $\Delta\boldsymbol{\phi}$. If the mapping $\Delta v^c \mapsto \Delta\boldsymbol{\phi}$ can be found, one can obtain $\Delta\boldsymbol{\phi}$ from Δv^c for gaze bias compensation.

To learn the mapping, we first collect training samples with different Δv^c via calibration. Here, we introduce a simple unconventional calibration method that captures a short video clip while the user is gazing at a fixed screen position and rotating his/her head freely. This process is very easy for a user to do and sufficient training samples ($\approx 10^2$) can be obtained in only a few seconds. Therefore, the training cost is quite low. For every obtained training sample $\{e_i, r_i, t_i, g_i\}$, we calculate the corresponding $\Delta\boldsymbol{\phi}_i$ and Δv_i^c, as mentioned above.

5.1.1. Calculation of $\Delta\boldsymbol{v}^c$ and $\Delta\boldsymbol{\phi}$

The camera's viewing direction vector v^c under HCS is determined by both head translation $t = [t^x, t^y, t^z]^T$ and head rotation $r = [r^x, r^y, r^z]^T$. Both t and r can be obtained by the head pose tracker in real time during calibration and estimation. Therefore, v^c can be calculated for both the calibration and estimation stages. The way to compute v^c from t and r is:

$$v^c = R\left(-t/\|t\|, r, [0,0,0]^T\right), \tag{8}$$

where function $R(\cdot)$ is defined by Eq. (16) .

Table 2
Comparison of estimation accuracy under a fixed head pose.

Method	Error	Training samples
Proposed	0.83°	33
Lu et al. [23]	0.62°	33
S^{3}GP + edge + filter [21]	0.83°	16 labeled and 75 unlabeled
Tan et al. [20]	0.5°	252
Baluja et al. [18]	1.5°	2000
Xu et al. [19]	1.5°	3000

The bias $\Delta\boldsymbol{\phi} = [\Delta\phi^x, \Delta\phi^y]$ rotates any initial gaze direction, namely, g^0, to the distortion compensated gaze direction denoted as g^d. Thus, $\Delta\boldsymbol{\phi} = [\Delta\phi^x, \Delta\phi^y]$ must fulfill the following relation.

$$g^d = \begin{pmatrix} \cos\Delta\phi^y & 0 & \sin\Delta\phi^y \\ 0 & 1 & 0 \\ -\sin\Delta\phi^y & 0 & \cos\Delta\phi^y \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\Delta\phi^x & -\sin\Delta\phi^x \\ 0 & \sin\Delta\phi^x & \cos\Delta\phi^x \end{pmatrix} g^0. \tag{9}$$

This problem can be solved to determine $\Delta\boldsymbol{\phi}$ as:

$$\begin{aligned} \Delta\phi^x &= \arctan\left(-\frac{g^{d,y}}{g^{d,z}}\right) - \arctan\left(-\frac{g^{0,y}}{g^{0,z}}\right), \\ \Delta\phi^y &= \arctan\left(\frac{g_i^{d,x}}{g^{d,z}}\right) + \arctan\left(\frac{g^{0,x}}{\left(1-(g^{0,x})^2-(g^{d,y})^2\right)^{\frac{1}{2}}}\right), \end{aligned} \tag{10}$$

where $g^{d,x}$ etc. are the elements of g^d, and $g^{0,x}$ etc. are the elements of g^0.

Note that g^d is supposed to be the distortion compensated gaze direction. In other words, it is our expected result after distortion compensation. Therefore, g^d is unknown during testing; we can only obtain it as training data during calibration stage. In particular, for the i-th training sample, we compute g_i^d by rotating g_i in accordance with head rotations r_i and r_0:

$$g_i^d = R(g_i, r_i, r_0), \tag{11}$$

which applies an inverse geometric compensation as described later in Section 5.2. The resulting g_i^d is then used to calculate $\Delta\boldsymbol{\phi}_i = [\Delta\phi_i^x, \Delta\phi_i^y]$ via Eq. (10) for the i-th training sample.

In Eq. (11), g_i is the ground truth gaze vector pointing from the user's eye to the known screen calibration position. Therefore, g_i depends on camera extrinsic parameters, i.e., relative position between the screen and camera, to convert head-camera coordinates into head-screen coordinates. If errors exist in these parameters, they will, however, have less effect on estimation accuracy. The reason is that as these parameters are fixed, they will cause similar gaze direction biases in both calibration and testing stages, and these biases partly counteract each other in testing.

5.1.2. Compensation via Gaussian process regression

After obtaining $\Delta\boldsymbol{\phi}_i$ and Δv_i^c for all training head poses, we learn the compensation for eye image distortion. The effect of geometric bias is removed first by a method introduced later in Section 5.2. Then, the mapping $\Delta v_i^c \mapsto \Delta\boldsymbol{\phi}_i$ is learnt by the Gaussian process regression (GPR) model. Note that $\{\Delta\boldsymbol{\phi}_i\} \in \mathbb{R}^2$ has two elements, so we learn two 1D regressions. If we take the first element $\{\Delta\phi_i^x\}$ as an example, the regression function is

$$\Delta\phi_i^x = f_x(\Delta v_i^c) \sim \mathcal{GP}\left(m(\Delta v_i^c), k_w\left(\Delta v_i^c, \Delta v_j^c\right)\right). \tag{12}$$

On the basis of the standard GPR model [31], we can define the terms in Eq. (12) and solve this problem as follows. First, the mean and covariance functions are defined by

$$\begin{aligned} & m(\Delta v_i^c) = 0, \\ & k_w\left(\Delta v_i^c, \Delta v_j^c\right) = k \exp\left(-\frac{\left\|\Delta v_i^c - \Delta v_j^c\right\|^2}{2l^2}\right) + \sigma^2\delta_{ij}, \end{aligned} \tag{13}$$

where σ^2 models the observation noise. For training, we write all the training data as $y = [\Delta\phi_1^x, \cdots, \Delta\phi_i^x, \cdots, \Delta\phi_n^x]^T$ and $\boldsymbol{V} = [\Delta v_1^c, \cdots, \Delta v_i^c, \cdots, \Delta v_n^c]^T$ and then optimize the hyperparameters $\omega = \{k, l, \sigma^2\}$ by minimizing the log marginal likelihood function:

$$\begin{aligned} \log p\left(y\middle|\boldsymbol{V}, \omega\right) = & -\frac{1}{2} y^T \left(\boldsymbol{K}_\omega(\boldsymbol{V}, \boldsymbol{V}) + \sigma^2 I\right)^{-1} y \\ & -\frac{1}{2}\log\left|\boldsymbol{K}_\omega(\boldsymbol{V}, \boldsymbol{V}) + \sigma^2 I\right| - \frac{n}{2}\log 2\pi, \end{aligned} \tag{14}$$

where $\boldsymbol{K}_\omega(\boldsymbol{V},\boldsymbol{V})$ is the covariance matrix whose element at (i,j) is $\Delta v_i^c, \Delta v_j^c$, shown in Eq. (13).

During estimation, for a test sample $\{\hat{e}, \hat{r}, \hat{t}\}$, $\Delta\hat{v}^c$ is computed from $\hat{r}$ and $\hat{t}$ first. Then, with the optimized hyperparameters, we predict $\Delta\hat{\phi}^x$ from $\Delta\hat{v}^c$ with

$$\begin{aligned} & \Delta\hat{\phi}^x = \boldsymbol{K}_\omega(\Delta\hat{\boldsymbol{v}}^c, \boldsymbol{V})\left(\boldsymbol{K}_\omega(\boldsymbol{V}, \boldsymbol{V}) + \sigma^2 I\right)^{-1} y, \\ & \mathrm{cov}\left(\Delta\hat{\phi}^x\right) = 1 - \boldsymbol{K}_\omega(\Delta\hat{\boldsymbol{v}}^c, \boldsymbol{V}) \\ & \qquad \left(\boldsymbol{K}_\omega(\boldsymbol{V}, \boldsymbol{V}) + \sigma^2 I\right)^{-1} \boldsymbol{K}_\omega(\boldsymbol{V}, \Delta\hat{\boldsymbol{v}}^c). \end{aligned} \tag{15}$$

After obtaining both $\Delta\hat{\phi}^x$ and $\Delta\hat{\phi}^y$, we use them to compensate for gaze estimation bias due to eye appearance distortion. This is done by rotating the initially estimated gaze vector $E_{r_0,t_0}(\hat{e}|T_0)$ in Section 4.2 around the X and Y axes with $\Delta\hat{\phi}^x$ and $\Delta\hat{\phi}^y$. The whole procedure is written as $E_{r_0,t_0}\left(\hat{e}|T_0\right) \otimes C_{r_0,t_0}^D\left(\hat{r}, \hat{t}\middle|T\right)$, following Eq. (3), where $C_{r_0,t_0}^D(\hat{r}, \hat{t}|T)$ represents the method described in this section.

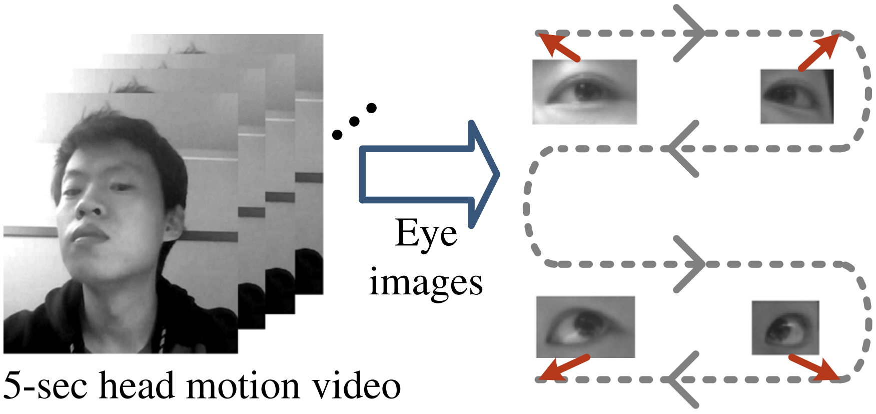

Fig. 9. Samples obtained from short video clip during head rotation.

Table 3
Ranges of camera's viewing direction of collected samples.

Rotation	Angle range
X-axis	−24.32°..24.75°
Y-axis	−27.48°..19.54°

5.2. Compensation 2: geometric bias

After compensating for eye appearance distortion, in this section, we handle the remaining gaze estimation bias due to geometric factors. In particular, because, until now, the gaze direction has been estimated by assuming a fixed head pose, (r_0, t_0), we need to further rotate it in accordance with the difference between r_0 and the real head orientation $\hat{r}$.

This problem can be defined as shown in Fig. 5. The original gaze vector computed for head orientation r_0 is known under WCS. Now, we want to rotate the HCS from $r_0 = [r_0^x, r_0^y, r_0^z]^T$ to $\hat{r} = \left[\hat{r}^x, \hat{r}^y, \hat{r}^z\right]^T$, while the gaze vector will undergo the same orientation. The question is, •How do we obtain the rotated gaze vector under WCS? Ž

We analyze the HCS rotation from r_0 to $\hat{r}$ step by step and apply these rotations to any vector, a_0, to obtain a. Note that here we use an arbitrary vector, a_0, rather than the gaze vector to discuss a general case because such rotations can be applied to any vector such as in Eq. (8) . The idea is to first rotate HCS from r_0 to $[0, 0, 0]^T$ and then further rotate it to $\hat{r}$. During each step, we apply rotations around the X, Y, and Z axes under the WCS in turn. In this manner, we can rotate any $a_0 \Rightarrow a$ by using the same rotations:

$$\begin{aligned} a &= R(a_0, r_0, \hat{r}) \\ &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_2^x & -\sin\theta_2^x \\ 0 & \sin\theta_2^x & \cos\theta_2^x \end{pmatrix} \begin{pmatrix} \cos\theta_2^y & 0 & \sin\theta_2^y \\ 0 & 1 & 0 \\ -\sin\theta_2^y & 0 & \cos\theta_2^y \end{pmatrix} \begin{pmatrix} \cos\theta_{12}^z & -\sin\theta_{12}^z & 0 \\ \sin\theta_{12}^z & \cos\theta_{12}^z & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ &\quad \begin{pmatrix} \cos\theta_1^y & 0 & \sin\theta_1^y \\ 0 & 1 & 0 \\ -\sin\theta_1^y & 0 & \cos\theta_1^y \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_1^x & -\sin\theta_1^x \\ 0 & \sin\theta_1^x & \cos\theta_1^x \end{pmatrix} a_0, \end{aligned} \tag{16}$$

where $\theta_1^x = -r_0^x$, $\theta_1^y = -\arctan(\tan r_0^y \cdot \cos r_0^x)$, $\theta_{12}^z = \hat{r}^z - r_0^z$, $\theta_2^y = \arctan\left(\tan \hat{r}^y \cdot \cos \hat{r}^x\right)$, and $\theta_2^x = \hat{r}^x$. By using Eq. (16) , we can complete the gaze estimation compensation described in Eq. (3) with

$$\begin{aligned} & E_{r_0,t_0}(\hat{e}|T_0) \otimes C^D_{r_0,t_0}(\hat{r}, \hat{t}|T) \otimes C^G_{r_0}(\hat{r}) \\ &= R(\underbrace{E_{r_0,t_0}(\hat{e}|T_0) \otimes C^D_{r_0,t_0}(\hat{r}, \hat{t}|T)}_{\text{Initial gaze + appearance distortion compensation}}, r_0, \hat{r}). \end{aligned} \tag{17}$$

This gives us the final gaze estimation result for any head pose, $\hat{r}, \hat{t}$.

Note that head translation $\hat{t}$ is not used in this section because in the current stage, we only focus on gaze direction; head translations $\hat{t}$ only shift eye/gaze positions but do not cause gaze directions to vary. If one wants to convert the finally obtained gaze direction into a gaze position on the screen, then $\hat{t}$ should be used to determine the gaze line's origin.

6. Experimental evaluations

The performance of the proposed method is evaluated by conducting several experiments. A gaze estimation system is built upon a desktop PC with a VGA resolution camera and a 22-inch LCD monitor. Users are required to sit in front of the monitor about 60 cm away. Then, gaze estimation experiments are performed in three stages: 1) collecting training samples with a fixed head pose, 2) collecting training samples with free head rotation by capturing a short video clip, and 3) performing gaze estimation tests with free head motion. Details of these procedures are explained in Section 3.3. Note that all gaze positions on screen are converted into gaze directions before they are used. This is done by first tracking 3D user eye positions via a head pose tracker and then computing the 3D directions from the eye positions to corresponding gaze positions on the screen.

Our experimental data is processed with non-optimized Matlab codes. The appearance distortion compensation (Gaussian process regression in estimation mode) and geometric calculation can run at 500 fps and >1000 fps, while the Lucas ..Kanade tracking, whose computational complexity is $O(n^2N + n^3)$ for N pixels and n alignment parameters [30] , becomes a bottleneck (<1 fps). However, using C/C++ for real-time Lucas ..Kanade implementations for larger problems has already been reported for many existing methods. Therefore, real-time implementation for our method is also possible.

6.1. Fixed-head pose gaze estimation

We first examine the fixed-head pose gaze estimation method described in Section 4.2 while the gaze compensation procedures are not involved. Training samples are collected by asking a user to gaze at each of the calibration points on screen, as shown in Fig. 8, with a fixed head pose. Then, test samples are also collected when a user freely chooses gaze positions while still keeping a fixed head pose. Six users are involved in the experiments, and their average estimation error is determined to be around 0.8°. A comparison between our result and those reported by previous fixed-head pose methods is shown in Table 2. Note that these methods all require personal calibration with a fixed head pose.

As shown in the table, our method achieves good accuracy among all methods. Although some other methods show a little better accuracy, they either need more training samples or implement much more complicated algorithms, while our method balances the tradeoff between accuracy and simplicity. Note that the aim of this paper is to handle free head motion; therefore, a simple and fairly good fixed-head pose estimator is preferred as a basis.

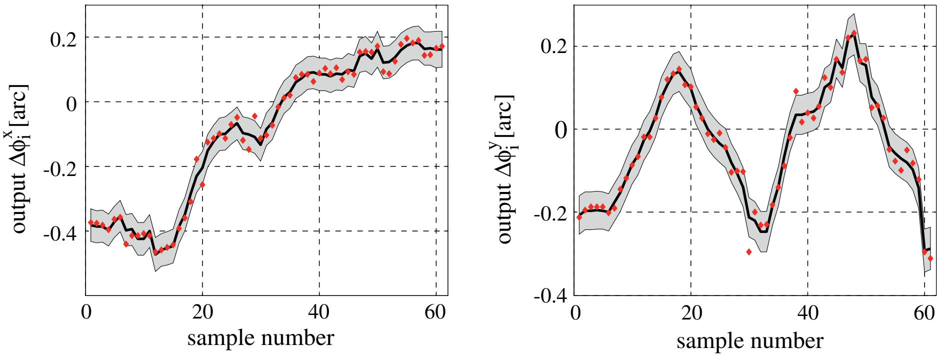

Fig. 10. Learnt regression model for $\Delta\phi_i^x$ and $\Delta\phi_i^y$. Dots indicate training samples' values, and lines indicate learnt results. Shaded region shows 90% confidence interval. Note similarity of regression results and head rotation trajectory in Fig. 9.

Table 4
Averages of gaze estimation errors with/without eye appearance distortion compensation.

Subject	With compensation	Without compensation
Subject 1	1.27 ± 0.79°	13.26 ± 7.41°
Subject 2	2.08 ± 1.34°	10.62 ± 7.06°
Subject 3	1.18 ± 0.83°	11.67 ± 5.03°
Subject 4	2.19 ± 1.36°	6.96 ± 5.79°
Subject 5	2.79 ± 1.76°	20.47 ± 11.44°
Subject 6	3.15 ± 1.80°	20.92 ± 11.49°
Subject 7	2.10 ± 1.86°	12.13 ± 6.07°
Average	2.11 ± 1.39°	13.72 ± 7.76°

Table 5
Head motion ranges under WCS in final experiments.

Head motion type	Range
X-translation	−93.6 mm ..93.0 mm
Y-translation	−5.7 mm ..72.2 mm
Z-translation	540.3 mm … 673.2 mm
X-rotation	−18.2°..19.4°
Y-rotation	−15.2°..16.0°
Z-rotation	−12.5°..10.7°

6.2. Eye appearance distortion compensation

We examine the eye appearance distortion compensation technique proposed in Section 5.1 quantitatively. For calibration, we show a static point in the center of the screen as the known gaze position. Then, every user is asked to fixate on that static point and rotate his/her head. At the same time, the camera captures the user's appearance in a 5-s video clip to obtain training samples for the experiment. Some captured eye images of these training samples are shown in Fig. 9, where the head rotation trajectory is roughly presented. Table 3 further gives the ranges of the camera's viewing direction variation due to head motion in X and Y rotations.

By using these training samples, we learn the Gaussian process regression model as described in Section 5.1. Fig. 10 shows the regression outputs for $\Delta\phi_i^x$ and $\Delta\phi_i^y$, which clearly present the relationships between eye appearance distortion compensation and head motion. In particular, $\Delta\phi_i^x$, which is the compensation angle around the X axis, increases with head rotation upward then downward, while the compensation angle $\Delta\phi_i^y$ around the Y axis alternately increases and decreases, reflecting the fact that head orientation varies between left and right.

To evaluate the model accuracy, we implement leave-one-out experiments by selecting each sample as a test sample and using the rest to train the regression model. Then appearance distortion compensation is applied to the test sample by using the trained model. Experimental results for all subjects are shown in Table 4, where the average gaze estimation error due to head motion reaches 13.7°, while it reduces to 2.1° after eye appearance distortion compensation via the learnt model. To intuitively show the error variation with head motion, per-image results for a representative subject are given in Fig. 11. Note that without compensation, the estimation errors become larger with significantly changed head poses, and they reduce when the head poses are close to the original one. However, with our compensation, the estimation errors are always stable and small. The significant difference in estimation errors with/without applying compensation demonstrates the effectiveness of the proposed method in compensating for eye appearance distortion.

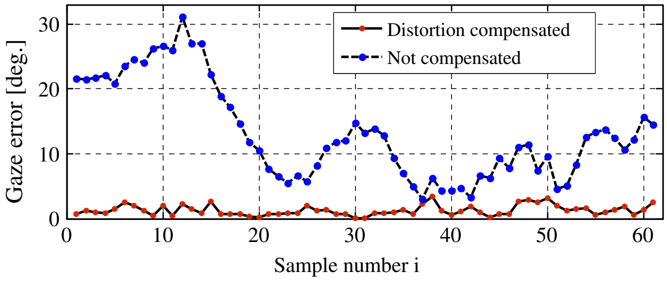

Fig. 11. Gaze estimation errors with/without appearance distortion compensation for a representative subject.

6.3. Estimation accuracy under free head motion

In this section we evaluate the gaze estimation accuracy of the proposed method under free user head motion. Experiments are done for ten subjects, and training samples are collected under a fixed-head pose and also from a short video clip, as described in the previous sections. During the test stage, each user chooses gaze positions on the screen randomly and uses a mouse click to inform the system about the gaze positions. Their head poses keep changing so that we can assess the ability of our method to handle head motion. The head motion ranges in the test stage of our experiments are provided in Table 5. The head translations in our experiments cover large ranges in which a user can freely translate his/her head without moving his/her body, while the head rotation ranges cover the full screen area. Therefore, the head motion ranges we tested are sufficient enough for common user ..computer scenarios.

We perform gaze estimation by using different compensation strategies for the collected test samples. For instance, Fig. 12 shows gaze estimation errors for subject S1, where results are obtained by applying full compensations, only geometric compensation, only distortion compensation, and no compensation. By comparing the errors, the effectiveness of the proposed compensation approach is clearly proven in terms of estimation accuracy, while using only one type of compensation or no compensation results in large estimation errors under user free head motion.

Furthermore, Fig. 13 plots the average gaze estimation errors for all ten subjects in the experiments. Note that the error values for different subjects may vary due to individuality and difference in head motion. However, the proposed method achieves the smallest estimation errors for all subjects only when both compensations are used. In particular, estimation accuracies of 2 ..3° are achieved by using our method under free head motion, while this number will be larger than 7.5° if head motion is not dealt with. This indicates the effectiveness of our method. Note that using only one type of compensation may cause larger errors than, when using one of them, as shown in Fig. 13. This is because eye appearance distortion and geometric factors produce gaze estimation

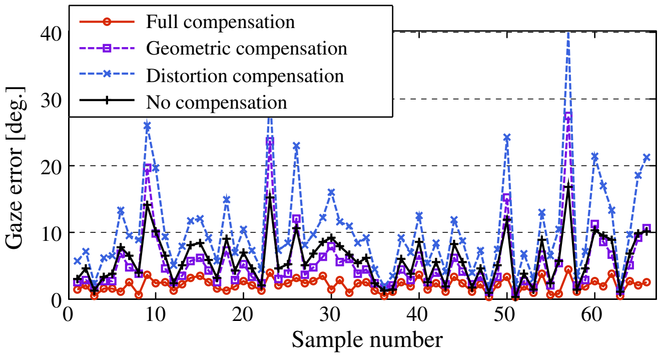

Fig. 12. Gaze estimation errors of all test samples from subject S1. Comparisons are shown with/without proposed compensations.

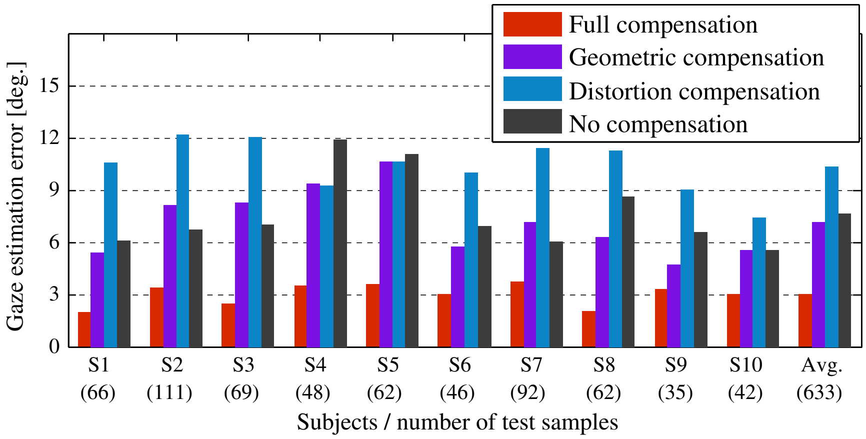

Fig. 13. Average gaze estimation errors for ten subjects with free head motion. Comparisons are made with/without proposed compensations. Numbers of test samples are given in parentheses.

errors in different ways. Their effects may counteract each other. Therefore, when only compensating for one of them, the remaining may be larger than before. In summary, the two compensations are both necessary in practice.

Finally, we quantitatively investigate how head tracking error affects our gaze estimation accuracy. The difficulty is that we do not have ground truth head poses, and thus, we do not know the head tracking errors. However, when only applying geometric compensation, increasing head pose error causes nearly the same gaze error increment. Therefore, we add Gaussian noise to the original head poses and only run geometric compensation. The increment of gaze error is 1.8° on average, which indicates a similar increment of head pose error. Then, we complete our test by using the same noisy head poses, and we get a gaze error increment of 0.7° for distortion compensation and 0.8° for full compensation. Such results show that head tracking error does not affect gaze estimation accuracy much. The two compensations appear to compensate for each other when head tracking error increases. In addition, the head pose tracker [29] used in our work reports an accuracy of 0.5..3°, which is already included in our final results.

A general comparison between our method and the previous ones that also allow for free head motion is given in Table 6. Besides using essentially different datasets, different methods also have specific requirements. For instance, the appearance-based methods need personal calibrations while the model-based methods need IR lights and cameras. Therefore we not only compare their reported accuracies but also listed their experimental conditions for a comprehensive comparison. From Table 6, we make the following observations.

1. Most existing gaze estimation methods that allow for head motion are model-based, and our method can be the state-of-the-art of the appearance-based methods. Its accuracy is comparable to the model-based methods.
2. The model-based methods usually require multiple IR lights and special cameras. Therefore, their usage is restricted.
3. Among the appearance-based methods, ours achieves both high accuracy and easy calibration. Compared with [28], which synthesizes eye images, our method is theoretically simpler and has less computational cost.

Regarding the public dataset, HPEG [32] is the most closely related dataset that contains both head pose and gaze information, and Heyman et al. [25] tested their method on HPEG. However, HPEG only categorizes gaze directions as •frontial Ž, •left Ž, •right Ž, etc., and it does not provide necessary training data. Therefore, it cannot be adopted by methods like ours for accurate gaze estimation and evaluation. Developing and using a comprehensive dataset for direct comparison are an important future work.

7. Conclusion and discussion

In this paper, we present a novel approach that performs appearance-based gaze estimation with free head motion. We solve this difficult problem by first decomposing it into subproblems and then solving them via initialization and compensations. Only a short video clip is required as an additional training input to allow for head motion. Experimental results show that our method achieves an average accuracy of around 3° by using only a single camera.

However, our method still has limitations. First, gaze estimation accuracy relies on head pose tracking accuracy, while our current head

Table 6
Comparison with previous methods that allow for head motion. Their reported accuracies and other characteristics are given.

Method	Category	Reported error	Camera(s)	Other requirements
Ours	Appearance	2..3°	1	Capture video
Lu et al. [28]	Appearance	2..3°	1	Eye image synthesis
Sugano et al. [27]	Appearance	4..5°	1	$\approx 10^3$ training samples
Nakazawa and Nitschke [9]	Model	0.9°	1 IR	IR LEDs & projector
Villanueva and Cabeza [8]	Model	1°	1 IR	2..4 IR LEDs
Zhu and Ji [7]	Model	2°	2 IR	n IR LEDs
Surveyed and summarized by [3]	Model	1..3°	1 IR	2 IR LEDs
	Model	1..2.5°	2 IR	4..5 IR LEDs
	Model	1..3°	2..4 IR	1..2 IR LEDs

pose tracker performs badly with large head rotation (>45°) and faces problems under varying illumination. Second, individual calibration needs to be done for every user before estimation. This is a common problem for appearance-based methods that is expected to be handled in the future.

Finally, the experimental validation in this paper uses our own dataset because existing dataset cannot fulfill the requirements for different methods. Therefore, a good goal for future work is to design and collect more general gaze estimation datasets for direct comparison between different methods.

References

[1] G. Underwood, Cognitive Processes in Eye Guidance, Oxford University Press, USA, 2005.

[2] L. Young, D. Sheena, Survey of eye movement recording methods, Behav. Res. Methods 7 (1975) 397–429.

[3] D. Hansen, Q. Ji, In the eye of the beholder: a survey of models for eyes and gaze, IEEE Trans. Pattern Anal. Mach. Intell. 32 (2010) 478–500.

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

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

[6] E. Guestrin, M. Eizenman, General theory of remote gaze estimation using the pupil center and corneal reflections, IEEE Trans. Biomed. Eng. 53 (2006) 1124–1133.

[7] Z. Zhu, Q. Ji, Novel eye gaze tracking techniques under natural head movement, IEEE Trans. Biomed. Eng. 54 (2007) 2246–2260.

[8] A. Villanueva, R. Cabeza, A novel gaze estimation system with one calibration point, IEEE Trans. Syst. Man Cybern. B Cybern. 38 (2008) 1123–1138.

[9] A. Nakazawa, C. Nitschke, Point of gaze estimation through corneal surface reflection in an active illumination environment, ECCV, 2012, pp. 159–172.

[10] R. Valenti, T. Gevers, Accurate eye center location and tracking using isophote curvature, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2008), 2008, pp. 1–8.

[11] R. Valenti, N. Sebe, T. Gevers, Combining head pose and eye location information for gaze estimation, IEEE Trans. Image Process. 21 (2012) 802–815.

[12] J. Wang, E. Sung, R. Venkateswarlu, Eye gaze estimation from a single image of one eye, Proceedings of the 9th IEEE International Conference on Computer Vision (ICCV 2009), 2003, pp. 136–143.

[13] D. Beymer, M. Flickner, Eye gaze tracking using an active stereo head, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2003), 2003, pp. 451–458.

[14] X. Brolly, J. Mulligan, Implicit calibration of a remote gaze tracker, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPRW 2004), 2004, p. 134.

[15] T. Nagamatsu, J. Kamahara, N. Tanaka, 3D gaze tracking with easy calibration using stereo cameras for robot and human communication, Proceedings of the 17th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN 2008), 2008, pp. 59–64.

[16] J. Kang, M. Eizenman, E. Guestrin, E. Eizenman, Investigation of the cross-ratios method for point-of-gaze estimation, IEEE Trans. Biomed. Eng. 55 (2008) 2293–2302.

[17] J. Chen, Q. Ji, Probabilistic gaze estimation without active personal calibration, CVPR, 2011, pp. 609–616.

[18] S. Baluja, D. Pomerleau, Non-intrusive gaze tracking using artificial neural networks, Proceedings of Advances in Neural Information Processing Systems, volume 6, 1994, pp. 753–760.

[19] L.Q. Xu, D. Machin, P. Sheppard, A novel approach to real-time non-intrusive gaze finding, Proceedings of British Machine Vision Conference (BMVC 1998), 1998, pp. 428–437.

[20] K. Tan, D. Kriegman, N. Ahuja, Appearance-based eye gaze estimation, Proceedings of the 6th IEEE Workshop on Applications of Computer Vision (WACV 2002), 2002, pp. 191–195.

[21] O. Williams, A. Blake, R. Cipolla, Sparse and semi-supervised visual mapping with the S^3GP, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2006), 2006, pp. 230–237.

[22] Y. Sugano, Y. Matsushita, Y. Sato, Calibration-free gaze sensing using saliency maps, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2010), 2010, pp. 2667–2674.

[23] F. Lu, Y. Sugano, T. Okabe, Y. Sato, Inferring human gaze from appearance via adaptive linear regression, Proceedings of the 13th IEEE International Conference on Computer Vision (ICCV 2011)2011.

[24] H. Yamazoe, A. Utsumi, T. Yonezawa, S. Abe, Remote gaze estimation with a single camera based on facial-feature tracking without special calibration actions, Proceedings of the 2008 Symposium on Eye Tracking Research and Applications, 2008, pp. 245–250.

[25] T. Heyman, V. Spruyt, L. Alessandro, 3d face tracking and gaze estimation using a monocular camera, Proceedings of the 2nd International Conference on Positioning and Context-Awareness2011.

[26] B.L. Nguyen, Y. Chahir, M. Molina, C. Tijus, F. Jouen, Eye gaze tracking with free head movements using a single camera, Proceedings of the 2010 Symposium on Information and Communication Technology, 2010, pp. 108–113.

[27] Y. Sugano, Y. Matsushita, Y. Sato, H. Koike, An incremental learning method for unconstrained gaze estimation, Proceedings of the 10th European Conference on Computer Vision (ECCV 2008), 2008, pp. 656–667.

[28] F. Lu, Y. Sugano, T. Okabe, Y. Sato, Head pose-free appearance-based gaze sensing via eye image synthesis, ICPR2012.

[29] faceAPI, http://www.seeingmachines.com/product/faceapi/ 2012.

[30] S. Baker, I. Matthews, Lucas–Kanade 20 years on: a unifying framework, IJCV 56 (2004) 221–255.

[31] C.E. Rasmussen, C.K.I. Williams, Gaussian Processes for Machine Learning, MIT Press, 2006.

[32] S. Asteriadis, D. Soufleros, K. Karpouzis, S. Kollias, A natural head pose and eye gaze dataset, Proceedings of the International Workshop on Affective-Aware Virtual Agents and Social Robots, 2009, pp. 1:1–1:4.