

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/265210917>

Adaptive Linear Regression for Appearance-Based Gaze Estimation

Article in IEEE Transactions on Pattern Analysis and Machine Intelligence · October 2014

DOI: 10.1109/TPAMI.2014.2313123

CITATIONS

170

READS

961

4 authors, including:



Feng Lu

Beihang University (BUAA)

97 PUBLICATIONS 1,847 CITATIONS

[SEE PROFILE](#)



Yoichi Sato

The University of Tokyo

312 PUBLICATIONS 9,461 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Gaze prediction [View project](#)



ACCV 2016: [View project](#)

Adaptive Linear Regression for Appearance-Based Gaze Estimation

Feng Lu, Yusuke Sugano, Takahiro Okabe, and Yoichi Sato

Abstract—We investigate the appearance-based gaze estimation problem, with respect to its essential difficulty in reducing the number of required training samples, and other practical issues such as slight head motion, image resolution variation, and eye blinking. We cast the problem as mapping high-dimensional eye image features to low-dimensional gaze positions, and propose an adaptive linear regression (ALR) method as the key to our solution. The ALR method adaptively selects an optimal set of sparsest training samples for the gaze estimation via ℓ^1 -optimization. In this sense, the number of required training samples is significantly reduced for high accuracy estimation. In addition, by adopting the basic ALR objective function, we integrate the gaze estimation, sub-pixel alignment and blink detection into a unified optimization framework. By solving these problems simultaneously, we successfully handle slight head motion, image resolution variation and eye blinking in appearance-based gaze estimation. We evaluated the proposed method by conducting experiments with multiple users and variant conditions to verify its effectiveness.

Index Terms—Eye, Gaze estimation, Face and gesture recognition, Sub-pixel alignment, Blink detection.



1 INTRODUCTION

EIGHTY percent of a human's sensory information is received by the eyes. Therefore, the ability to track human gaze direction is essential for use in an intelligent system. For instance, it has long been suggested that gaze trackers could be a good alternative to traditional input devices for computers, while some commercial systems have already been developed for people with disabilities so they can accomplish tasks by only gazing. Other possible applications can be found in various areas including human computer interaction, marketing research, virtual training, and medical research.

Computer vision-based gaze estimation technology predicts human gaze by capturing and analyzing the eye images. The currently existing methods can be classified into two categories: model-based and appearance-based. Methods in the former category appear earlier in literature. Basically, they involve geometric models of the eyeball and the environment, and extract small eye features to fit the model. The most commonly extracted features include corneal infrared reflections [1], [2], pupil center [3], and iris contour [4]. The gaze direction is computed using these features along with a wide variety of methods from conventional eyeball modeling [5] to a cross-ratios-based geometric method [6], [7]. However, for accurate extraction of small eye features, their systems usually involve NIR imaging devices, stereo camera pairs, and pan-tilt/long-focus cameras [8], [9], [10], [11], and can be quite difficult to build and calibrate. Comprehensive surveys can be found in [12], [13].

Unlike feature-based methods, appearance-based methods use an entire eye image as a high-dimensional input feature and map this feature to low-dimensional gaze position space. Their advantage lies in that as long as there is no need to extract small scale eye features, it is sufficient enough to use only a single web camera with a relative low resolution instead of requiring a complex system setup.

On the other hand, appearance-based methods have their disadvantages. In particular, obtaining an accurate mapping needs a large number of training samples. Baluja and Pomerleau [14] proposed a neural network and collected 2000 samples for training. A similar method was also introduced by Xu *et al.* [15]. Tan *et al.* [16] investigated the local linearity in the appearance feature manifold and interpolated unknown gaze position using 252 training samples. Clearly, obtaining such a great number of training samples via tedious calibration procedures is unacceptable by ordinary users.

Williams *et al.* [17] introduced a semi-supervised Gaussian Process regression method to reduce the number of labeled training samples. However, many unlabeled samples are still required. Sugano *et al.* [18] introduced a unique approach that combines gaze estimation with saliency extraction. However, it is reported to have a relatively low estimation accuracy. Other recent methods were proposed by using new image features and different regression techniques [19], [20], [21].

Some other limitations for appearance-based methods come with the fact that the gaze estimation accuracy is seriously affected by eye appearance changes. For instance, free head motion significantly deforms eye appearance, and thus it is regarded as another major problem in appearance-based gaze sensing that still remains challenging to deal with [22], [23], [24]. In addition, some other factors also greatly influence the

• Feng Lu, Yusuke Sugano, Takahiro Okabe, and Yoichi Sato are with the Institute of Industrial Science, the University of Tokyo, Tokyo 153-8505, Japan.
E-mail: {lufeng, sugano, takahiro, ysato}@iis.u-tokyo.ac.jp

observed eye appearances, such as misalignment of the eye regions and eye blinking. These issues should be carefully treated in practical gaze sensing.

This paper extends our previous work [25]. It has been already adopted by [26] and [27] to allow for subject-independent calibration and free head motion, respectively, at the price of reduced accuracy and using additional inputs (RGBD images). In this paper, we aim at a comprehensive solution that solves multiple limitations in appearance-based gaze estimation without requiring additional inputs, while it maintains a high accuracy and also has the good extensibility.

Contribution. We take into account the key problems in appearance-based gaze estimation. First, as the most essential question, is it possible to perform accurate gaze estimation using only a small number of training samples for easy calibration? Second, if allowing free head motion is too difficult, what can we do to support slight head motion as a tradeoff? Finally, we also investigate two other important issues in practical gaze estimation, namely image resolution variation and blinking, to see how they influence the estimation accuracy and how to handle their effects.

We propose a novel adaptive linear regression (ALR) method based on an ℓ^1 -optimization framework to solve the above problems. It utilizes sparse and low-dimensional training samples to predict gaze positions from the input eye images. In this sense, the number of required training samples can be significantly reduced. Moreover, under the same optimization framework, we deal with slight head motion via an accurate sub-pixel alignment. Finally, we propose how to solve the problems with image resolution variation and blinking. All these problems are solved under a unified framework.

The following characteristics explicitly help distinguish our work from others:

- 1) An ALR method is proposed that optimally selects a small number of training samples for estimation (Section 2). Therefore, it only requires sparsely collected training samples with low training costs.
- 2) A sub-pixel alignment method is proposed that generalizes the basic optimization framework of ALR. The alignment and gaze estimation are done simultaneously to handle the problems due to slight head motion and image resolution variation (Section 3 and Section 4).
- 3) A blink detection method is proposed within the same optimization framework. It detects blinks while not being disturbed by gaze changes. It works for every single image without requiring information from the neighboring frames (Section 4).

2 GAZE ESTIMATION VIA ADAPTIVE LINEAR REGRESSION

We propose an adaptive linear regression method aiming at reducing the number of training samples while maintaining a good gaze estimation accuracy.

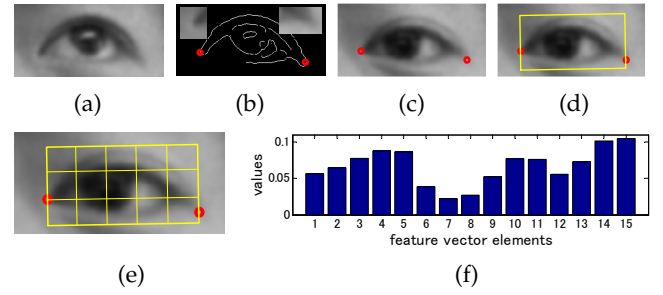


Fig. 1: **Eye appearance feature extraction.** Above: typical eye region alignment process. (a) One eye image used as anchor. (b) Two corners of the eye are detected by edge filter for anchor eye image. The image intensities around the two corners of the eye are saved as templates. (c), (d) For any other eye images, eye corners are detected via template matching, and then alignment region is determined. Bottom: (e) Illustration of 3×5 subregion division, and (f) generated 15-D feature vector.

2.1 Low-dimensional feature extraction

Existing appearance-based methods generate the eye image feature $\mathbf{e}_i \in \mathbb{R}^m$ from i -th captured image \mathbf{I}_i by raster scanning all its pixels, thus the typical feature dimensionality m reaches several thousand [16], [15] or even higher (e.g., edge map is added in [17]). On one hand, a high-dimensional feature keeps all the information in the image. On the other hand, since gaze directions have only two degrees of freedom, such high-dimensional features are highly redundant. Moreover, actually captured eye regions can be of variant resolutions, and therefore, pixel-wise feature extraction faces the problem of inconsistent output dimensions.

We use a low-dimensional feature extraction method consisting of two steps: eye region alignment and feature generation. In the first step, the eye regions are accurately aligned for different eye images. This is commonly done as illustrated in Fig. 1(a)-(d), where the inner and outer eye corners from an anchor eye image are detected by an edge filter, and then the eye corner regions are stored as image templates (Fig. 1(b)) to match the eye corners of other eye images (Fig. 1(c)) and finally align the eye region \mathbf{I}_i (Fig. 1(d)).

The above eye image alignment is used based on the observation that the relative positions and appearances of human eye corners remain almost unchanged [28]. However, as described later in Section 3.2, such feature point-based alignment may be insufficient in practice. Partly for this reason, many existing appearance-based methods require a fixed head pose using a chinrest, limiting their applicability significantly.

In the feature generation step, once the aligned eye image region \mathbf{I}_i is obtained, it is further divided into $p \times q$ even subregions, as shown in Fig. 1(e). Let S_j denote the summation of the pixel intensities in the j -th subregion, then the feature vector is generated as $\mathbf{e}_i = (\sum_j S_j)^{-1} [S_1, S_2, \dots, S_{p \times q}]^T$ (Fig. 1(f)). Note that

both the division and intensity summation are taken with the sub-pixel accuracy. That is, the edges of the divided regions may freely cross any pixel grid, and the summation $S_j = \sum_k r_k i_k$ takes weighted pixel intensities where the weights $\{r_k\}$ represent the ratios of the occupied areas by the subregion on the pixel grids.

2.2 Linearity in eye appearance manifolds

All the eye appearance features $\{e_i\} \in \mathbb{R}^m$, which are extracted from the accurately aligned image regions, constitute a manifold in the m -D space. Since the eyeball movement has only two degrees of freedom, the manifold has an intrinsic dimensionality of close to two. To test this statement, we project all the features onto a 3-D space by PCA for visualization, as shown in Fig. 2. Several observations can be made here. **First, the eye feature manifold can be approximated as a 2-D surface with most of its information accumulating inside the first two major dimensions.** Second, the proposed 15-D feature well keeps more information in the first three major dimensions (Fig. 2(c)) than the pixel-wise extracted feature (Fig. 2(b)). **Therefore, although the manifold in Fig. 2(b) seems smoother, the information can be hidden in other dimensions.** Finally but most importantly, the projected features on the manifolds in Fig. 2(b) and (c) show a similar pattern to the gaze positions in Fig. 2(a).

These observations help us better understand the efficiency of the linear interpolation-based methods [22], [16]. Their basic idea is to reconstruct the test appearance feature \hat{e} by using a linear combination of the training features, which are denoted as $\{e_i\}$.

$$\hat{e} = \sum_i w_i e_i \quad s.t. \quad \sum_i w_i = 1 \quad (1)$$

Eq. (1) solves the weights $\{w_i\}$. Then by letting $\{x_i\}$ denote the corresponding gaze positions of the training samples, one can calculate the gaze position \hat{x} of the test sample by:

$$\hat{x} = \sum_i w_i x_i \quad (2)$$

For those weights $\{w_j \neq 0\}$ computed by Eq. (1), **denote their corresponding eye features and gaze positions as $\{e'_j\}$ and $\{x'_j\}$, respectively.** Then, because Eq. (2) uses the same $\{w_i\}$ as from Eq. (1), these methods in fact assume that the linear combinations in the subspaces spanned by $\{e'_j\}$ and $\{x'_j\}$ are equal.

Since $\{e'_j\}$ are from a manifold in a high-dimensional space, without any prior knowledge, **the above method is valid only by assuming locality, i.e., limiting $\{e'_j\}$ within a sufficient small region centered by \hat{e} in the manifold.** By assuming local linearity in the manifold, one can expect the relationships between Eq. (1) and Eq. (2) to be approximately held.

The existing methods [22], [16] guarantee this locality assumption by obtaining dense training samples, from which $\{e'_j\}$ are selected with the smallest Euclidean distances from \hat{e} . However, if the training samples are only

sparsely collected, as shown in Fig. 2, the local linearity assumption cannot be satisfied. Therefore, problem with sparse training samples motivates our methods. **Our idea is to adaptively find an optimal set of training samples that best reconstruct the test image linearly,** and we show that by using the same linear combination, gaze estimation can be done accurately without requiring dense training data.

In practice, there can be additional problems caused by head motion, illumination change and eye opening degrees. For head motion and its resulting illumination change, because our method assumes a fixed head pose like most appearance-based methods, their effect can be handled. Although we further allow for slight head motion later in this paper. For variation in eye opening degree, we propose a blinking detection method to handle it. Finally, also like most appearance-based methods, our method should be trained for different people to learn individual adaptive mappings.

2.3 Adaptive linear regression (ALR)

Let matrices $E = [e_1, e_2, \dots, e_n] \in \mathbb{R}^{m \times n}$ and $X = [x_1, x_2, \dots, x_n] \in \mathbb{R}^{2 \times n}$ consist of all the eye features and gaze positions of the training samples. A simple linear regression from E to X can be obtained by finding a transformation:

$$AE = X \quad (3)$$

where $A \in \mathbb{R}^{2 \times m}$ is the projection matrix. When training samples' number $n > m$, Eq. (3) is overdetermined and cannot be exactly satisfied by any A . In another words, we cannot find a global linear regression that is accurate for all the n training samples.

However, if we find a linear mapping for only a subset $\{(e'_j, x'_j)\}$ that contains $n' < m$ training samples by:

$$A'E' = X' \quad (4)$$

where $E' = [e'_1, \dots, e'_{n'}] \in \mathbb{R}^{m \times n'}$ and $X' = [x'_1, \dots, x'_{n'}] \in \mathbb{R}^{2 \times n'}$, then certain A' can be found that accurately maps e'_j to x'_j .

Clearly, A' can only perform accurate mapping for the training samples in the subset $\{(e'_j, x'_j)\}$. However, for an arbitrarily input test sample (\hat{e}, \hat{x}) , there is no guarantee that $A'\hat{e} = \hat{x}$. **Then the idea is that we can choose the optimal subset $\{(e'_j, x'_j)\}$ with respect to \hat{e} , so that the resulting A' is more probably accurate for estimating \hat{x} from \hat{e} .** In this sense, we independently pursue the best mapping for every test sample.

Intuitively, \hat{e} should be closely correlated to $\{e'_j\}$ so that they tend to share the same linear mapping. We seek such *correlation* by selecting the fewest e'_j that can still interpolate \hat{e} linearly with total weight equals to one. Thus, \hat{e} is considered to live in the low-dimensional subspace spanned by $\{e'_j\}$ and probably shares the same A' in the estimation. Therefore, we calculate:

$$\hat{w} = \arg \min_w \|w\|_0 \quad s.t. \quad Ew = \hat{e} \in \mathbb{R}^m, \mathbf{1}^T w = 1 \quad (5)$$

15特征值将更多的信息保存在前两个维度

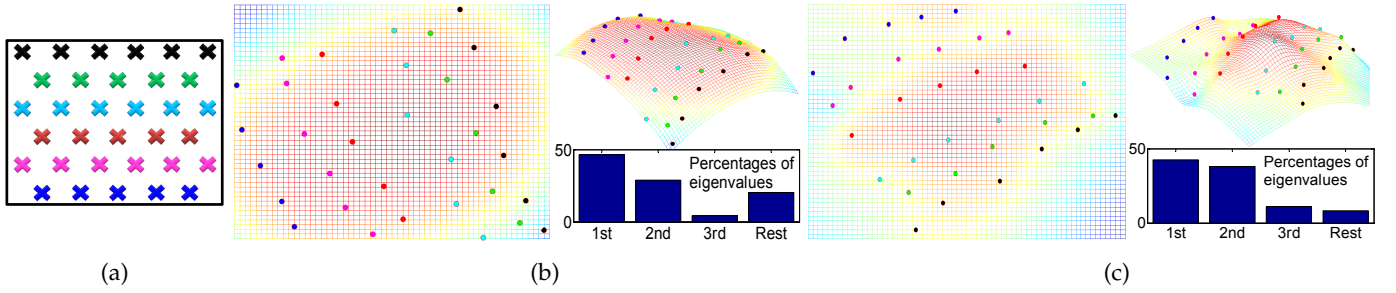


Fig. 2: **2-D gaze space and eye appearance feature manifold.** (a) Illustration of 33 gaze positions on 2-D screen. (b) Projection of corresponding eye appearance manifold on 3-D space. The magnitudes of the eigenvalues are shown as percentages. (c) Similarly, illustration of manifold projection for our proposed 15-D low dimensional feature. Notice the similarity between the gaze positions on the screen and the feature coordinates on the manifold.

where $\|\cdot\|_0$ indicates the ℓ^0 -norm. Then, $\{e'_j\}$ is selected from $\{e_i\}$ in accordance with the non-zero weights in $\hat{\mathbf{w}}$. Finally, the gaze position is estimated from $\hat{\mathbf{e}}$ by:

$$\begin{aligned}\hat{\mathbf{x}} &= \mathbf{A}'\hat{\mathbf{e}} = \mathbf{A}'\mathbf{E}\hat{\mathbf{w}} = \mathbf{A}'\mathbf{E}'\hat{\mathbf{w}}' + \mathbf{A}'\mathbf{E}^0\hat{\mathbf{w}}^0 \\ &= \mathbf{X}'\hat{\mathbf{w}}' + \mathbf{0} = \mathbf{X}'\hat{\mathbf{w}}' + \mathbf{X}^0\hat{\mathbf{w}}^0 = \mathbf{X}\hat{\mathbf{w}} \in \mathbb{R}^2\end{aligned}\quad (6)$$

where $\hat{\mathbf{w}}'/\hat{\mathbf{w}}^0$ is composed of the non-zero/zero elements in $\hat{\mathbf{w}}$, and $\mathbf{E}^0/\mathbf{X}^0$ is formed by the columns from \mathbf{E}/\mathbf{X} that correspond to $\hat{\mathbf{w}}^0$.

With Eq. (6), we only need to solve Eq. (5) for $\hat{\mathbf{w}}$ without explicitly obtaining \mathbf{A}' . However, solving the problem in Eq. (5) is NP-hard [29]. Fortunately, an ℓ^1 -norm minimization is an alternative when $\hat{\mathbf{w}}$ is sparse enough [30], [31]. At the same time, note that according to Section 2.1, any extracted feature e_i satisfies $\mathbf{1}^T e_i = 1$. Thus from $\mathbf{E}\mathbf{w} = \hat{\mathbf{e}}$, we have

$$\mathbf{1}^T(\mathbf{E}\mathbf{w}) = \mathbf{1}^T(\hat{\mathbf{e}}) \Rightarrow [\cdots, \mathbf{1}^T e_i, \cdots] \mathbf{w} = 1 \Rightarrow \mathbf{1}^T \mathbf{w} = 1 \quad (7)$$

which naturally ensures the second constraint in Eq. (5) and thus this constraint can be removed. In addition, a tolerance term ε should be introduced to make a tradeoff between the sparsity and linear combination precision:

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} \|\mathbf{w}\|_1 \quad s.t. \quad \|\mathbf{E}\mathbf{w} - \hat{\mathbf{e}}\|_2 < \varepsilon \quad (8)$$

By minimizing $\|\mathbf{w}\|_1$, we are actually pursuing the lowest dimensional subspace where the interpolated feature $\mathbf{E}\mathbf{w}$ lives. The value of ε restricts the maximum allowed Euclidean distance from $\mathbf{E}\mathbf{w}$ to the true $\hat{\mathbf{e}}$. Thus, with a small ε , $\hat{\mathbf{e}}$ is considered to be approximately in the same subspace, and the local transformation in Eq. (4) is applicable. In addition, as ε is sufficiently small, Eq. (7) can still hold true. As an important complement to the proposed method, the optimal choice of ε is thoroughly investigated later in Section 5.1 along with experimental demonstrations.

Relationship to other works. ℓ^1 -minimization has been used in previous vision-related researches, among which the works by Wright *et al.* [32], Wagner *et al.* [33] and Tan *et al.* [34] share similarities to ours. However, our work essentially differs in some aspects. First, they apply face recognition, which is a classification problem, while

we handle a typical regression problem. In this sense, our focus is not discriminability but accurate estimation, and we novelly show the effectiveness of ℓ^1 -optimization-based method in handling a regression problem. Second, Wright *et al.* [32] also introduce an error term ε without mentioning how to determine its value. However, we demonstrate later in Section 5.1 that the ε value is crucial to our solution and should be carefully chosen, while it is not the case for classification problems. Another difference is that [32] assumes sparsity in reconstruction errors while we assume fewer supporting training samples for optimal representation for a query image.

3 HANDLING SLIGHT HEAD MOTION

The basic ALR method described in Section 2 solves the problem with a fixed head pose. However, in practice head motion is always inevitable, which significantly affects the estimation accuracy. Since allowing for free head motion remains a challenge, we instead take into consideration slight head motion.

3.1 Slight head motion

We refer to slight head motion as the inevitable and unintentional user head translation or rotation at small amplitudes. As appearance-based gaze estimation under free head motion remains a great challenge, the proposed ALR method in Section 2 focuses on the fixed head pose case. However, it is not easy for an ordinary user to keep his/her head absolutely stable, and no one would like to use a chinrest in everyday life. Therefore, it is necessary to study how and how much the slight head motion affects the gaze estimation accuracy, in the case that a user naturally keeps a relatively stable head pose without needing additional equipment.

We make two observations about the gaze estimation under slight head motion. First, slight head motion affects the estimated gaze position by two factors: 1) *geometric factor* geometrically varies the gaze direction in accordance with the head pose; and 2) *image factor* affects the estimation by deforming the eye images.

Second, the way the slight head motion deforms a captured eye image can be sufficiently approximated by

a rigid transformation, i.e., 2D-translation, and rotation inside the image plane, and other types of complex photometric distortions can be ignored. This is understandable when taking into consideration the imaging formula for a camera and the human face geometry (approximated by a plane), on the condition that the head motion is relatively small compared to the distance between the user's face and the camera.

It is beneficial to first discuss how the two factors affect the gaze estimation accuracy and which of them is dominant following the first observation. On one hand, if we only take into account geometric factor, the gaze direction bias will obviously have the same magnitude as the head motion itself. Therefore, if the head motion is slight, the resulting gaze variation is small. On the other hand, if we take into account image factor, the gaze estimation error caused by eye image deformation can be dozens of times larger (Appendix A). Therefore, the image factor, which takes the form of the eye image deformation, is the major cause of gaze estimation error.

Subsequently, it is natural to believe that if we can correct the eye image deformation, the gaze estimation accuracy under slight head motion will be effectively improved. As rigid transformation is used to model the eye image deformation, the correction can be done via eye image alignment, which finds the correct rigid transformation and then performs the inverse transformation.

In conclusion, for slight head motion, one can perform eye image alignment via 2D rigid transformation to improve the gaze estimation accuracy.

3.2 Sub-pixel eye region alignment

This section proposes an eye image alignment method. The method works with the sub-pixel accuracy because the resolution of the captured eye regions is always limited. Moreover, it should have the ability to align eye images with different resolutions.

Conventional feature point-based alignment methods face difficulties including: 1) with low resolution, it is difficult to accurately find feature points such as the corners of the eye with sub-pixel accuracy, and 2) the features may be unstable under different resolutions. Therefore, we use the entire eye in our alignment.

In addition, typical alignment methods warp the image during alignment, which results in expensive computational costs. However, following the feature extraction method in Section 2.1, we propose transforming the extraction regions and compute the feature directly with sub-pixel accuracy, as shown in Fig. 3(a)(b). In this way, the computational cost is significantly reduced.

We compute the optimal *rigid transformation* for the extraction regions according to Section 3.1. The feature extraction process is formulated as:

$$e_\tau = f(\mathbf{I}, \mathbf{p}) = f(\mathbf{I}, \tau(\mathbf{p}_0)) \quad (9)$$

where $f(\cdot)$ is the feature extraction function, \mathbf{I} is the captured image, \mathbf{p}_0 is the pixel coordinates of the initial

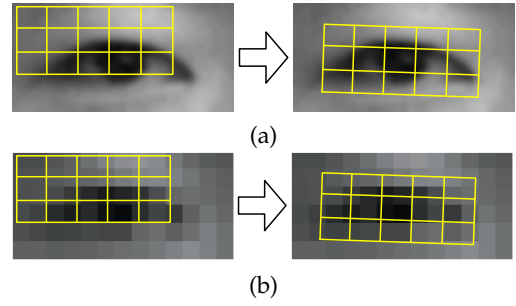


Fig. 3: **Illustration of feature extraction region alignment.** (a) Optimal rigid transformation is found to accurately locate extraction regions. (b) Alignment at low resolution.

extraction regions' vertices and \mathbf{p} is the final extraction regions' vertices after a rigid transformation $\tau(\cdot)$:

$$\mathbf{p} = \tau(\mathbf{p}_0) = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \mathbf{p}_0 + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (10)$$

Therefore, our target is to find the optimal $\tau(\cdot)$, which is denoted as $\hat{\tau}(\cdot)$, to generate the optimal feature $e_{\hat{\tau}}$.

It is insufficient to align the input eye image to any of the single training eye images in order to find the optimal $\tau(\cdot)$. Instead, we propose an optimization method under the same framework described in Section 2.3.

Recall that $\|\mathbf{w}\|_1$ in Eq. (8) can be sufficiently minimized only when the input feature lives in the low-dimensional manifold consisting of all the training data. Such low-dimensionality corresponds to the intrinsic 2-D eyeball rotation. On the other hand, misalignment of the eye region introduces extra degrees of freedom and the generated feature lies outside the manifold. Therefore, it is reasonable to assume that accurate alignment minimizes Eq. (8). Based on this observation, we seek the optimal alignment function $\hat{\tau}(\cdot)$ by minimizing the ALR function in Eq. (8) with respect to $\tau(\cdot)$.

Fig. 4 shows an example of how the minimized $\|\hat{\mathbf{w}}\|_1$ varies for different alignments of the extraction regions by plotting densely computed results. Since there are three parameters in a rigid transformation including x-translation, y-translation, and rotation, we vary them for two combinations of x/y-translations and x-translation/rotation for easy 2-D visualization. Fig. 4(a) shows the results for a high resolution case. The size of the entire extraction region is 150×75 pixels, and the ranges of the parameter variations are $20 \sim 130$ pixels for both the x/y-translations (from the top-left corner of the original image) and $-45^\circ \sim 45^\circ$ for the rotation. There are clearly global minimal values for $\|\hat{\mathbf{w}}\|_1$ corresponding to the accurate alignment. Furthermore, Fig. 4(b) shows a low resolution case where the test eye image undergoes a 1/10 down-sampling and the extraction region size becomes 15×7.5 pixels. However, the variation of $\|\hat{\mathbf{w}}\|_1$ appears to be nearly the same as that in the high resolution case, and thus, still leads to the accurate alignment with a sub-pixel precision.

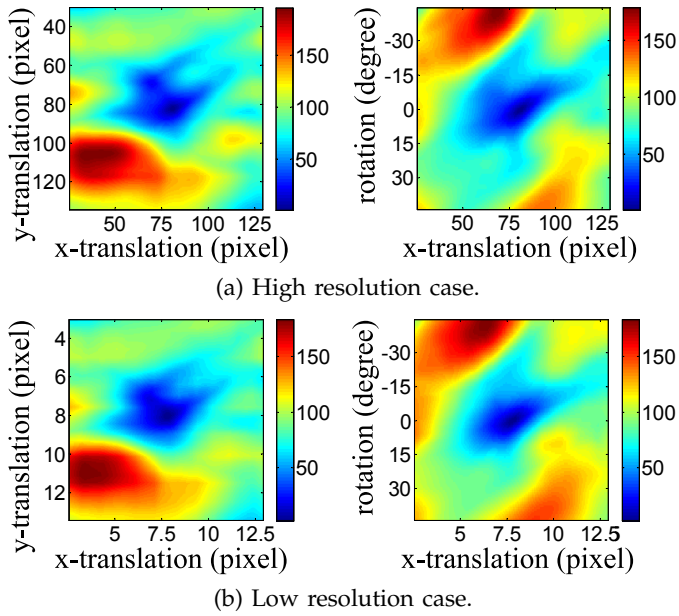


Fig. 4: **Relationship between $\|\hat{\mathbf{w}}\|_1$ and alignment.** The variation in $\|\hat{\mathbf{w}}\|_1$ due to x/y-translations of the alignment region is shown on the left, and that due to x-translation/rotation is shown on the right. The color bars display the values of $\|\hat{\mathbf{w}}\|_1$.

Algorithm 1 (Alignment with ALR)

Input:

training image features E
test image I
initial alignment regions vertices \mathbf{p}_0

Output:

optimal linear combination weights $\hat{\mathbf{w}}$
optimal rigid transformation $\hat{\tau}(\cdot)$

- 1: **while** not converged **do**
 - 2: extract feature $e_\tau = f(I, \tau(\mathbf{p}_0))$
 - 3: update $\hat{\mathbf{w}} \leftarrow \arg \min_{\mathbf{w}} \|\mathbf{w}\|_1$ s.t. $\|E\mathbf{w} - e_\tau\|_2 < \varepsilon$
 - 4: compute¹ $\nabla \|\mathbf{w}\|_1|_{\hat{\mathbf{w}}} = [\frac{\partial \|\mathbf{w}\|_1}{\partial t_x}, \frac{\partial \|\mathbf{w}\|_1}{\partial t_y}, \frac{\partial \|\mathbf{w}\|_1}{\partial \alpha}]|_{\hat{\mathbf{w}}}$
 - 5: compute² $\Delta\tau = (\nabla \|\mathbf{w}\|_1|_{\hat{\mathbf{w}}})^{-1}(1 - \|\hat{\mathbf{w}}\|_1)$
 - 6: update $\tau(\cdot)$ by $\Delta\tau$
 - 7: **end while**
 - 8: **return** optimal solution $\hat{\mathbf{w}}, \hat{\tau}(\cdot)$
-

The results from Fig. 4 suggest a joint optimization for both the alignment $\tau(\cdot)$ and gaze estimation \mathbf{w} . This problem can then be formulated based on the basic ALR:

$$(\hat{\tau}(\cdot), \hat{\mathbf{w}}) = \arg \min_{\mathbf{w}, \tau(\cdot)} \|\mathbf{w}\|_1 \quad \text{s.t.} \quad \|E\mathbf{w} - e_\tau\|_2 < \varepsilon \quad (11)$$

where e_τ is computed by using Eq. (9).

Simultaneously solving both $\tau(\cdot)$ and \mathbf{w} is difficult. Instead, we update each of them respectively by fixing the other for every iteration. In particular, \mathbf{w} can be updated by solving the ℓ^1 -minimization problem; while we compute $\Delta\tau$ instead of $\tau(\cdot)$ with a linearization treatment to update $\tau(\cdot)$.

As described later in Section 5.1, the expected value of the optimal $\|\hat{\mathbf{w}}\|_1$ is 1. Therefore, if we linearize $\|\mathbf{w}\|_1$ at the current $\tau(\cdot)$ via first order approximation, one can expect $\|\mathbf{w}\|_1 + \nabla \|\mathbf{w}\|_1 \cdot \Delta\tau \rightarrow 1$ and compute $\Delta\tau$ by using Newton's method. Then, $\tau(\cdot)$ is updated by $\Delta\tau$. As a result, the problem can be rewritten for each iteration as

$$(\hat{\Delta\tau}, \hat{\mathbf{w}}) = \arg \min_{\mathbf{w}, \Delta\tau} \|\mathbf{w}\|_1 \quad (12)$$

$$\text{s.t. } \|E\mathbf{w} - e_\tau\|_2 < \varepsilon, \|\mathbf{w}\|_1 + \nabla \|\mathbf{w}\|_1 \cdot \Delta\tau - 1 = 0$$

where $\nabla \|\mathbf{w}\|_1$ contains the derivatives of $\|\mathbf{w}\|_1$ w.r.t. the parameters in $\tau(\cdot)$, and $\Delta\tau = [\Delta t_x, \Delta t_y, \Delta\alpha]^T$ consists of the increments of the parameters in $\tau(\cdot)$.

The detailed method is summarized in Algorithm 1. For initialization, one can use any simple eye detector or other methods to roughly locate eye positions. As shown in Fig. 4, a global minimum that corresponds to the optimal alignment solution should be easily achieved after a roughly correct initialization (in Fig. 4: ± 50 pixels in translations and $\pm 25^\circ$ in rotation) without being trapped by a local minimum. Then, the proposed method simultaneously performs the gaze estimation and eye image alignment to improve both results.

Relationship to other works. Wagner *et al.* [33] also proposed using ℓ^1 -minimization for alignment with regard to the face recognition problem. However, their method minimizes the ℓ^1 -norm of the reconstruction errors for robustness, while our method minimizes the ℓ^1 -norm of reconstruction weights for an accurate regression. Thus, they are essentially different in idea. Peng *et al.* [35] handled the alignment problem for a set of images using RPCA, which shares an inherent similarity with ℓ^1 -minimization. However, all the images under their circumstance can be transformed without taking into consideration the different roles of the training/test samples as in our case. Furthermore, our alignment method works with regression jointly, which is another important feature.

4 TWO MORE PRACTICAL ISSUES

In addition to head motion, other problems exist that affect practical gaze estimation. In this section, we discuss two important issues including the change of the eye image resolution and eye blinking. We analyze how they affect the gaze estimation results, and solve the resulting problems by using techniques built upon the basic ALR method with a uniform optimization framework.

4.1 Change of eye image resolution

Under a controlled environment in the laboratory, it is not necessary to change the capture resolution during

1. $\nabla \|\mathbf{w}\|_1|_{\hat{\mathbf{w}}}$ in Step 4 is computed numerically by varying each parameter in $\tau(\cdot)$ along both its positive and negative directions and examining the difference in the resulting $\|\mathbf{w}\|_1$. The variation step-length is set to be half of $\Delta\tau$ in the last iteration.

2. $(\nabla \|\mathbf{w}\|_1|_{\hat{\mathbf{w}}})^{-1}$ in Step 5 computes the inverse of $\nabla \|\mathbf{w}\|_1|_{\hat{\mathbf{w}}}$ by $\frac{1}{3}[\nabla_x^{-1}, \nabla_y^{-1}, \nabla_\alpha^{-1}]|_{\hat{\mathbf{w}}}$, where ∇_x , ∇_y , and ∇_α are the three elements of $\nabla \|\mathbf{w}\|_1$.

gaze sensing. However, it may be required for applications in the future. On the one hand, the training step is always important, and thus, the training eye images will be captured in high resolution to guarantee accuracy. On the other hand, the test images may be captured at a lower resolution in some applications to save the system resources (e.g., memory, bandwidth, available CPU times), because gaze estimation may be introduced only as a dependent component of other main applications.

The key to supporting resolution variation is making sure the eye features extracted from different resolution eye images are uniform. This is guaranteed by the feature extraction method in Section 2.1. As shown in Fig. 1(e), the eye features are computed with the same predefined extraction regions to have a fixed low-dimension. The size of the regions can be easily determined using the same image resolution scaling factor. In addition, the calculation is set to sub-pixel precision so that the eye features can be accurately extracted from the variant or even extremely low resolution images, and then, indifferently sent to the gaze estimator. In this way, the proposed gaze estimation method naturally supports eye image resolution variation.

Another essential requirement is that the eye regions should be accurately aligned before the feature extraction for different resolution eye images. In particular, the alignment for low resolution eye images is very difficult for feature point-based methods. Our method solves this problem by using the proposed sub-pixel alignment method, which is described in Section 3.2.

4.2 Eye blink

Eye blink is another issue to consider. In the literature of eye/gaze tracking, blinking has always been used as a tool for eye localization [36] or remote operation [37], while has merely been considered as a negative influential factor. However, in practice, its negative effect is worthy of attention.

4.2.1 Problem

An eye image captured during blinking appears different from a normal image. Therefore, the appearance-based gaze estimator creates a sudden change in the continuously estimated gaze positions, which is incorrect. Moreover, because eye blinks happen frequently although transitorily, the ceaseless changes in the estimated gaze positions result in unacceptable instability and prevent the user from completing the task.

A straightforward way to stabilize the estimated gaze trajectory is to apply smoothness filtering. However, the gaze directions tend to change abruptly, e.g., gaze change in saccade. The common gaze change pattern consists of two phases: 1) fixing on an interesting point for a period of time and 2) quickly jumping to another fixation point in a discontinuous way. If we want to use the smoothness filter, we need to determine whether the estimated gaze variation is caused by eye blinks or real gaze changes,

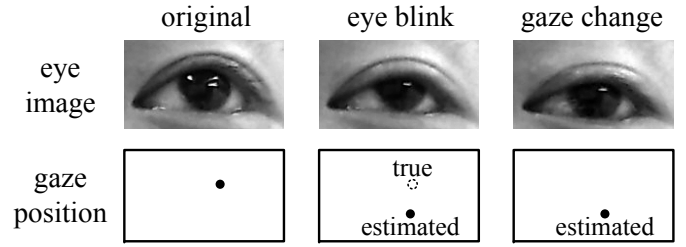


Fig. 5: Example of estimated gaze variations due to eye blinks and gaze change. Real gaze change (right) correctly moves the estimated gaze position. On the other hand, eye blinking (middle) may also result in a similar estimated gaze position, while such gaze variation is not correct because the true gaze position should be the same as the original one. Conventional appearance-based gaze estimators cannot distinguish between these two cases, and thus, cannot remove the errors due to eye blinking.

and then perform the filtering only in the first case while preserving the reasonable discontinuity in the second one. However, there is no explicit clue to distinguish them in real time. An example is shown in Fig. 5.

4.2.2 Blink detection

In this section, we propose an eye detection method that automatically determines whether a sudden estimated gaze variation is caused by a real gaze change or a blink. The key idea is that, $\|\mathbf{w}\|_1$ in Eq. (8) can be sufficiently minimized (the expected value is 1, see Section 5.1) only when the input eye feature lives in the low-dimensional manifold consisting of training data. The low-dimensionality corresponds to the intrinsic 2-D eyeball rotation. On the other hand, eye blinking doesn't change the eyeball orientation, but produces unseen blinking eye features. This introduce extra degrees of freedom and the blinking eye features must live outside the manifold, meaning that the expected minimum value cannot be achieved in Eq. (8).

Motivated by the above observation, our method measures the eye condition and detects eye blinks by checking the minimized $\|\hat{\mathbf{w}}\|_1$ in ALR (Eq. (8)). In particular, a value close to 1 implies a non-blinking eye. Otherwise, the eye image is considered captured in an abnormal state (e.g., during a blink).

This method is also built upon the basic ALR. It detects blinks without requiring any information from the previous/next frame, and thus, helps to correct and stabilize the estimated gaze trajectory in real time.

Advantages over other methods. The majority of blink detection methods assume that the pixel intensity change between temporally successive frames indicates eye blinking [36], [38]. Some methods further examine the optical flow in the eye regions [39], [40], while others may model eye appearance changes via AAM [41] or by using iris/eyelid contours [42]. However, as shown in the experiments later, such eye appearance differences may also be caused by gaze changes, which their meth-

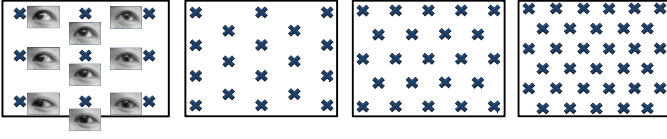


Fig. 6: **Patterns of calibration points in training.** The crosses indicate the calibration points on the screen during training. The users should focus his/her gaze on each of the points in turn to capture the training eye images. Four different patterns were designed with totals of 9, 18, 23, and 33 calibration points in each.

ods cannot distinguish. On the other hand, our method detects blinking from only a single image without temporally neighboring frames.

Overall, the ability of our method to detect abnormal eye images comes naturally. The key is that the proposed ALR method is designed by investigating the underlying structure of the data, *i.e.*, the low-dimensional structure of the specific high-dimensional observations. Therefore, any invalid sample, no matter how it is produced (misalignment, blink, *etc.*), can be easily recognized if it conflicts with the underlying data structure.

5 EXPERIMENTAL EVALUATION

We evaluated the performance of the proposed method via extensive experimentation. To obtain the training and test data, we implemented our system on a desktop PC with a 22-inch LCD monitor and a webcam whose resolution can go up to 1280×1024 . In the experiment, the user was asked to sit in front of the monitor (about 50 ~ 60 cm away) and keep his/her head stable with or without the help of a chinrest. Then, we conducted the following training and test stages.

- 1) **Training stage.** The user focused his/her gaze on each calibration point shown on the screen and allowed the camera to capture his/her frontal appearance. Then, positions of the calibration points were saved and the eye image features were extracted. Training should be done for every user.
- 2) **Test stage.** The user was shown random test points one by one on the screen and the camera again captured the user's appearances for gathering the test eye image features.

The details for each specific experiment are further explained in the following sections. Moreover, since one of our goals is to allow for a sparse sampling of the training data, we designed four sampling patterns on the screen containing only 9, 18, 23, or 33 calibration points, as shown in Fig. 6.

Estimation error is calculated by angular difference to measure the accuracy:

$$\text{error} \approx \arctan\left(\frac{\|\hat{\mathbf{x}} - \mathbf{x}'\|_2 \cdot \cos \theta}{D / \cos \theta}\right) \quad (13)$$

where $\|\hat{\mathbf{x}} - \mathbf{x}'\|_2$ denotes the Euclidean distance between a real 2D gaze position \mathbf{x}' and the estimated 2D gaze

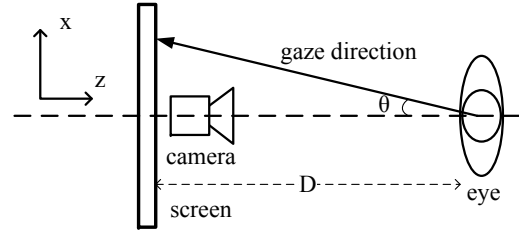


Fig. 7: **Vertical view of gaze estimation system.**

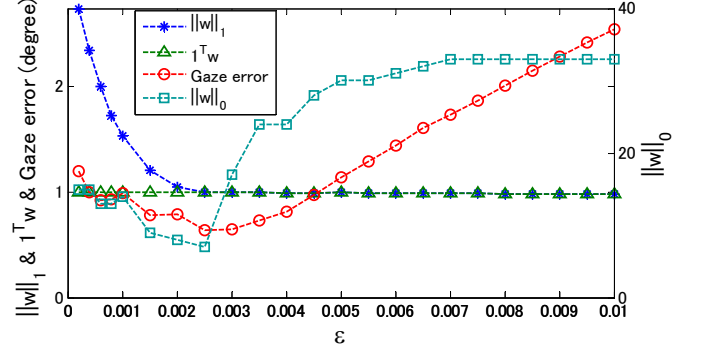


Fig. 8: **Example of solution variation with different ε .** With ε increasing, $\|\hat{\mathbf{w}}\|_1$ is reduced, while the values of $\|\hat{\mathbf{w}}\|_0$ and the gaze estimation error only decrease at first and then begins to increase. The best ε choice occurs when $\|\hat{\mathbf{w}}\|_1$ just converges to 1.

position $\hat{\mathbf{x}}$, D indicates the distance between the user's eye and the screen, and θ is the angle between the gaze direction and z -axis, as shown in Fig. 7.

5.1 Determining ε for ℓ^1 -optimization

This section completes the discussion on determining the value for ε following Section 2.3 with an experimental demonstration. Intuitively, larger ε results in smaller $\|\hat{\mathbf{w}}\|_1$ from Eq. (8). Then, the essential issues include how is the proper value of ε determined? What is more important, how does the variation of ε affect the minimization of $\|\hat{\mathbf{w}}\|_0$?

Fig. 8 shows a typical experimental result from solving the problem with different ε for one test sample. Clearly, with ε increasing, the value of $\|\hat{\mathbf{w}}\|_1$ keeps reducing. However, the $\|\hat{\mathbf{w}}\|_0$ value only decreases at first and then begins to increase, while the gaze estimation error varies in the same way. Note that the minimum value of $\|\hat{\mathbf{w}}\|_0$ is reached exactly when $\|\hat{\mathbf{w}}\|_1$ converges to $\mathbf{1}^T \hat{\mathbf{w}} = 1$.

This phenomenon is interesting while understandable. Note that if Eq. (7) holds true, then

$$\|\mathbf{w}\|_1 \geq \mathbf{1}^T \mathbf{w} = 1 \quad (14)$$

meaning $\|\hat{\mathbf{w}}\|_1$ reaches the minimum of 1 when all the elements in $\hat{\mathbf{w}}$ become non-negative. After this, $\|\hat{\mathbf{w}}\|_1$ equals $\mathbf{1}^T \hat{\mathbf{w}}$, thus further increasing ε turns Eq. (8) into

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} (\mathbf{1}^T \mathbf{w}) \quad s.t. \quad \|\mathbf{E}\mathbf{w} - \hat{\mathbf{e}}\|_2 < \varepsilon \quad (15)$$

Similar to the derivation in Eq. (7), a constraint for $\mathbf{1}^T \hat{\mathbf{w}}$ can be found:

$$\mathbf{1}^T \hat{\mathbf{w}} = \mathbf{1}^T (\mathbf{E}\mathbf{w} - \hat{\mathbf{e}}) + 1 \quad (16)$$

which further converts the optimization into:

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} (\mathbf{1}^T (\mathbf{E}\mathbf{w} - \hat{\mathbf{e}})) \quad s.t. \quad \|\mathbf{E}\mathbf{w} - \hat{\mathbf{e}}\|_2 < \varepsilon \quad (17)$$

Clearly, Eq. (17) no longer minimizes $\|\hat{\mathbf{w}}\|_0$. Instead, it makes $\hat{\mathbf{w}}$ dense and thus $\|\hat{\mathbf{w}}\|_0$ increases. Moreover, since Eq. (17) minimizes $\mathbf{1}^T \hat{\mathbf{w}}$, the constraint of $\mathbf{1}^T \mathbf{w} = 1$ begins to break down. Due to these reasons, the original problem in Eq. (5) can no longer be deal with.

Therefore, the optimal ε value should be determined when the minimized $\|\hat{\mathbf{w}}\|_1$ converges to 1, so that: 1) such an ε achieves the sparsest solution $\hat{\mathbf{w}}$, which is obtained by using Eq. (8) rather than Eq. (17), while not breaking the constraint of $\mathbf{1}^T \mathbf{w} = 1$; and 2) the non-negativity of $\hat{\mathbf{w}}$ ensures that the linear combination is performed inside the convex region constituted by $\{\mathbf{e}'_j\}$, which is a good property for interpolation.

An easy way to find the best ε is to iteratively solve the problem in Eq. (8) while gradually increasing ε until $\|\hat{\mathbf{w}}\|_1 = 1$. On the other hand, doing this for every test sample is time consuming. Based on the observation that the optimal ε value is very stable for the same dataset, we can use a constant value ε_c for all the test samples:

$$\varepsilon_c = \frac{\sum_i \alpha_i \varepsilon_i}{\sum_i \alpha_i}, \quad \alpha_i = \exp(-\kappa \|\mathbf{x}_i - \hat{\mathbf{x}}\|_2) \quad (18)$$

where ε_i is the optimal choice for estimating the i -th training sample from the rests, and $\hat{\mathbf{x}}$ is the screen center position. With a positive κ , the calculated weights $\{\alpha_i\}$ in Eq. (18) ensure that ε_i for the peripheral sample is less influential to ε_c . The reason for this is that the ε_i for the peripheral sample tends to be larger, as illustrated and explained in Fig. 9. We empirically found that Eq. (18) works well in most cases unless the gaze positions are in the peripheral regions of the display. Therefore, we decided to use the constant ε_c due to its simplicity.

5.2 Evaluation and comparison for basic ALR

We evaluate the estimation accuracy of the basic ALR method in this section. Details of the experimental setups and approaches include:

- 1) **Training samples.** As shown in Fig. 6, the training samples were sparsely collected in four sets with totals of 9, 18, 23, and 33.
- 2) **Test samples.** For each training set, nearly 100 test samples were collected whose gaze positions were randomly chosen on the screen.
- 3) **Dataset size.** The four sets of training/test samples were collected for each of the seven subjects.
- 4) **Fixed head pose.** A chinrest was used to help stabilize the users' heads.
- 5) **Eye image alignment.** With fixed head poses, the eye regions were directly cropped from the same region for all images.

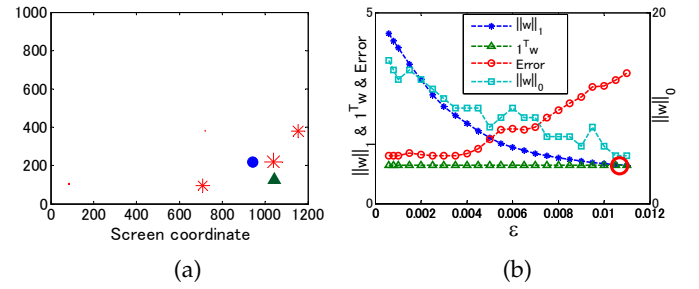


Fig. 9: Example of estimating peripheral gaze position. (a) True gaze position (triangle) lies outside of the region composed of the supporting samples (stars), and therefore, the linear reconstruction with all-nonnegative weights cannot be accurate (circle). Instead, an extrapolation would be better with negative weights. (b) Estimation results with varying ε . As explained above, it is better to allow for negative weights. Thus, the best ε should be chosen before the $\|\hat{\mathbf{w}}\|_1$ converging to 1.

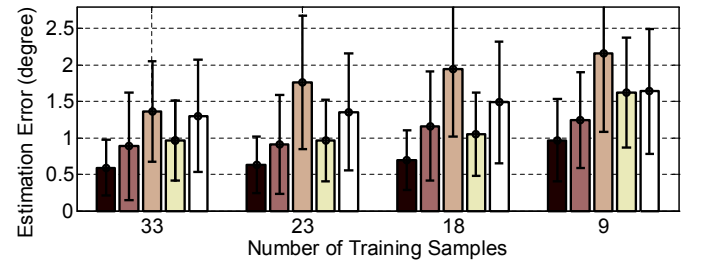


Fig. 10: Comparison of average gaze estimation results in Table 1. For each number of training samples, results of ALR, Local region [16], PCA+GPR [19], HOG+SVR [20], and CSLBP+GPR [21] are shown from left to right.

- 6) **Feature.** In our case, 15-D features were extracted with 3×5 subregions described in Section 2.1. While for other methods in comparison, different features were generated and used as they proposed.

We compare the proposed ALR method with some latest appearance-based methods in terms of gaze position estimation accuracy. These methods were proposed by using different feature descriptors as and regression techniques [16], [19], [20], [21]. We use the same dataset to test their performances. As a result, their estimation errors are given in Table 1. The proposed method shows the highest estimation accuracy in different experimental conditions. In general, it achieved estimation accuracies of better than 1° . As for other methods, although they use more complex descriptors and different regression techniques, their accuracies are clearly not as good as ours in the conditions of sparse training samples. For better visualization of the comparison, Fig. 10 plots the average estimation errors and their standard deviations to show the advantages of the proposed method.

As described in Section 2, the proposed ALR method benefits the estimation by adaptively choosing the supporting training samples in an optimization manner. This is briefly demonstrated via two individual estimation

TABLE 1: Comparison of gaze estimation error (in degree) between proposed ALR method and other methods using same datasets. The dataset was collected for seven subjects. For each subject, four different training sets were obtained for a total of 33, 23, 18, and 9 training samples. For each training sample set, the average gaze estimation result was computed using nearly 100 randomly collected test samples.

Training samples	Method	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5	Subject 6	Subject 7	Average
33	Proposed ALR	$0.49 \pm 0.28^\circ$	$0.54 \pm 0.42^\circ$	$0.67 \pm 0.44^\circ$	$0.58 \pm 0.40^\circ$	$0.62 \pm 0.39^\circ$	$0.57 \pm 0.34^\circ$	$0.66 \pm 0.38^\circ$	$0.59 \pm 0.38^\circ$
	Local region [16]	$0.57 \pm 0.37^\circ$	$0.70 \pm 0.46^\circ$	$1.37 \pm 1.40^\circ$	$0.99 \pm 0.85^\circ$	$0.97 \pm 0.74^\circ$	$0.79 \pm 0.80^\circ$	$0.79 \pm 0.53^\circ$	$0.88 \pm 0.73^\circ$
	PCA+GPR [19]	$1.39 \pm 0.71^\circ$	$1.42 \pm 0.68^\circ$	$1.21 \pm 0.70^\circ$	$1.59 \pm 0.86^\circ$	$1.01 \pm 0.56^\circ$	$1.83 \pm 0.70^\circ$	$1.09 \pm 0.62^\circ$	$1.36 \pm 0.69^\circ$
	HOG+SVR [20]	$0.84 \pm 0.54^\circ$	$0.88 \pm 0.49^\circ$	$0.86 \pm 0.48^\circ$	$1.03 \pm 0.62^\circ$	$0.95 \pm 0.48^\circ$	$1.01 \pm 0.66^\circ$	$1.17 \pm 0.58^\circ$	$0.96 \pm 0.55^\circ$
	CSLBP+GPR [21]	$1.28 \pm 0.75^\circ$	$1.18 \pm 0.65^\circ$	$1.13 \pm 0.67^\circ$	$1.25 \pm 0.67^\circ$	$1.28 \pm 0.85^\circ$	$1.34 \pm 0.84^\circ$	$1.64 \pm 0.96^\circ$	$1.30 \pm 0.77^\circ$
23	Proposed ALR	$0.53 \pm 0.33^\circ$	$0.51 \pm 0.32^\circ$	$0.66 \pm 0.38^\circ$	$0.75 \pm 0.51^\circ$	$0.67 \pm 0.42^\circ$	$0.58 \pm 0.33^\circ$	$0.69 \pm 0.43^\circ$	$0.63 \pm 0.39^\circ$
	Local region [16]	$0.84 \pm 0.57^\circ$	$0.72 \pm 0.46^\circ$	$0.93 \pm 0.57^\circ$	$0.98 \pm 0.76^\circ$	$1.10 \pm 0.98^\circ$	$0.84 \pm 0.76^\circ$	$0.97 \pm 0.65^\circ$	$0.91 \pm 0.68^\circ$
	PCA+GPR [19]	$1.82 \pm 0.97^\circ$	$1.26 \pm 0.53^\circ$	$1.93 \pm 0.92^\circ$	$2.52 \pm 1.57^\circ$	$1.50 \pm 0.75^\circ$	$2.06 \pm 0.96^\circ$	$1.23 \pm 0.70^\circ$	$1.76 \pm 0.92^\circ$
	HOG+SVR [20]	$0.94 \pm 0.51^\circ$	$0.80 \pm 0.45^\circ$	$0.88 \pm 0.46^\circ$	$1.09 \pm 0.70^\circ$	$0.98 \pm 0.55^\circ$	$1.01 \pm 0.62^\circ$	$1.07 \pm 0.62^\circ$	$0.97 \pm 0.56^\circ$
	CSLBP+GPR [21]	$1.35 \pm 0.96^\circ$	$1.12 \pm 0.61^\circ$	$1.24 \pm 0.67^\circ$	$1.28 \pm 0.76^\circ$	$1.43 \pm 0.85^\circ$	$1.49 \pm 0.87^\circ$	$1.59 \pm 0.87^\circ$	$1.36 \pm 0.80^\circ$
23	Proposed ALR	$0.55 \pm 0.37^\circ$	$0.51 \pm 0.33^\circ$	$0.80 \pm 0.51^\circ$	$0.69 \pm 0.41^\circ$	$0.76 \pm 0.38^\circ$	$0.76 \pm 0.46^\circ$	$0.78 \pm 0.42^\circ$	$0.69 \pm 0.41^\circ$
	Local region [16]	$0.72 \pm 0.53^\circ$	$0.91 \pm 0.54^\circ$	$1.55 \pm 1.04^\circ$	$1.02 \pm 0.58^\circ$	$1.48 \pm 0.93^\circ$	$1.06 \pm 0.63^\circ$	$1.38 \pm 0.99^\circ$	$1.16 \pm 0.75^\circ$
	PCA+GPR [19]	$1.81 \pm 0.71^\circ$	$1.45 \pm 0.83^\circ$	$1.57 \pm 0.68^\circ$	$2.73 \pm 1.44^\circ$	$2.26 \pm 0.96^\circ$	$2.37 \pm 1.15^\circ$	$1.45 \pm 0.78^\circ$	$1.95 \pm 0.93^\circ$
	HOG+SVR [20]	$0.87 \pm 0.52^\circ$	$0.95 \pm 0.52^\circ$	$1.02 \pm 0.53^\circ$	$1.02 \pm 0.58^\circ$	$1.19 \pm 0.55^\circ$	$1.27 \pm 0.73^\circ$	$1.01 \pm 0.58^\circ$	$1.05 \pm 0.57^\circ$
	CSLBP+GPR [21]	$1.39 \pm 0.75^\circ$	$1.41 \pm 0.74^\circ$	$1.24 \pm 0.75^\circ$	$1.21 \pm 0.68^\circ$	$1.79 \pm 1.01^\circ$	$1.72 \pm 0.98^\circ$	$1.65 \pm 0.92^\circ$	$1.49 \pm 0.83^\circ$
23	Proposed ALR	$0.69 \pm 0.44^\circ$	$0.71 \pm 0.34^\circ$	$1.30 \pm 0.74^\circ$	$0.83 \pm 0.69^\circ$	$1.32 \pm 0.70^\circ$	$0.97 \pm 0.51^\circ$	$0.97 \pm 0.56^\circ$	$0.97 \pm 0.57^\circ$
	Local region [16]	$0.77 \pm 0.45^\circ$	$1.12 \pm 0.46^\circ$	$1.30 \pm 0.71^\circ$	$1.56 \pm 0.83^\circ$	$1.45 \pm 0.81^\circ$	$1.16 \pm 0.67^\circ$	$1.31 \pm 0.66^\circ$	$1.24 \pm 0.66^\circ$
	PCA+GPR [19]	$2.18 \pm 1.18^\circ$	$1.80 \pm 0.74^\circ$	$2.14 \pm 1.08^\circ$	$2.76 \pm 1.20^\circ$	$2.29 \pm 1.19^\circ$	$2.73 \pm 1.38^\circ$	$1.23 \pm 0.76^\circ$	$2.16 \pm 1.08^\circ$
	HOG+SVR [20]	$1.67 \pm 0.72^\circ$	$1.43 \pm 0.72^\circ$	$1.46 \pm 0.75^\circ$	$2.01 \pm 0.76^\circ$	$1.63 \pm 0.78^\circ$	$1.66 \pm 0.71^\circ$	$1.50 \pm 0.81^\circ$	$1.62 \pm 0.75^\circ$
	CSLBP+GPR [21]	$1.76 \pm 1.10^\circ$	$1.31 \pm 0.67^\circ$	$1.51 \pm 0.80^\circ$	$1.24 \pm 0.70^\circ$	$2.06 \pm 0.89^\circ$	$1.99 \pm 0.89^\circ$	$1.60 \pm 0.97^\circ$	$1.64 \pm 0.86^\circ$

examples in Fig. 11. It is clear that the ALR method adaptively selected fewer supporting samples among all the training samples and achieved a better level of estimation accuracy. On the other hand, the Local region-based method used all neighboring training samples for minimizing the reconstruction error, while the estimation error was not minimized. In summary, our method outperforms others in optimally selecting training samples to guarantee the estimation accuracy, in the case of sparse training samples.

Moreover, Table 2 compares our method with other early appearance-based methods based on their reported results obtained with different training data. Although our method used much fewer training samples, its accuracy is still very high. Using 15-D features and 18 training samples is a good tradeoff between easy calibration and high precision. Other notable points include: 1) these existing methods use high dimensional features generated from high resolution eye images, while our method only extracts very low dimensional features regardless of the eye image resolution; and 2) our method adaptively selects supporting training samples, and thus, the system is very flexible in terms of directly adding or removing any training samples into or out of the system.

The computational cost simply depends on the ℓ^1 -solver. In our experiments we use L1-magic [43] in Matlab, and it runs with a 2.8GHz CPU and 8G memory in real time (28fps). For extension methods which require iterations in the next section, their computational costs will increase linearly. In their cases, real time estimation

TABLE 2: Comparison of our method with other existing appearance-based methods based on their reported estimation errors and number of training samples.

Method	Error	Training samples	Dimensionality
Proposed	0.97°	9	15
Proposed	0.69°	18	15
Proposed	0.59°	33	15
S ³ GP [17]	1.32°	16 labeled and 75 unlabeled	$10^4 \sim 10^5$
S ³ GP+edge+filter [17]	0.83°	16 labeled and 75 unlabeled	$10^4 \sim 10^5$
Tan <i>et al.</i> [16]	0.5°	252	4000
Baluja <i>et al.</i> [14]	1.5°	2000	500 \sim 800
Xu <i>et al.</i> [15]	1.5°	3000	600

has not been achieved via pure matlab codes. As an important future work, using optimized C++ codes to achieve real time implementation is possible.

5.3 Evaluation with additional factors

In this section, we take into consideration the practical issues including slight head motion, low eye image resolution, and eye blinking.

5.3.1 Evaluation with slight head motion

The restriction of a fixed head pose is relaxed in this section. In particular, we evaluated the method proposed in Section 3 for handling slight head motion. A total of

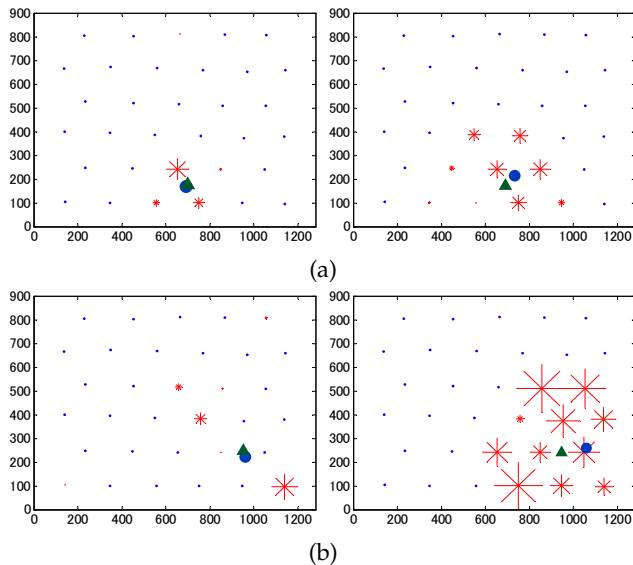


Fig. 11: Two estimation examples. Left figures show our estimation results and right ones come from the Local region method [16]. Circles and triangles represent the estimated gaze positions and their ground truth in the screen coordinates, and the stars are the selected training samples with their sizes indicating the absolute values of linear combination weights. Note that [16] produced both positive and negative weights with large absolute values, although their summation was still 1.

eight subjects were involved in this experiment. For each person, he/she was asked to sit in front of the screen about 50 ~ 60 cm away and try to keep a natural and fixed head pose *without* a chinrest. The training and test stages were done as follows:

- 1) **Training stage.** As shown in Fig. 6, 23 training samples were collected while the user gazed at each of the 23 calibration points shown on the screen.
- 2) **Test stage.** For each user, nearly 100 test samples were collected whose gaze positions were randomly shown on the screen.
- 3) **Head pose.** The users were asked to keep a natural and fixed head pose without a chinrest, while slight head motion (several degrees) was unavoidable.

We evaluated our method by comparing the resulting gaze estimation errors for the following methods:

- 1) **Proposed method.** Described in Section 3. Eye positions were initialized by roughly locating the eye position via any simple eye detector.
- 2) **Direct estimation.** Direct gaze estimation via basic ALR by ignoring slight head motion. This method is expected to provide the baseline results.
- 3) **Face alignment.** Handling head motion by using a state-of-the-art commercial head pose tracker [44]. Training/test eye images were cropped via face alignment. This method represents the most commonly adopted solution for such a problem.

Experiments were done on the same obtained dataset. The results are given in Table 3. Not surprisingly, a

TABLE 3: Gaze estimation with slight head motion.

	Proposed method	Direct estimation	Face alignment	Head motion [mm]
S1	$1.49 \pm 0.97^\circ$	$10.18 \pm 7.44^\circ$	$4.00 \pm 2.52^\circ$	4.7 ± 2.6
S2	$1.69 \pm 0.98^\circ$	$12.79 \pm 8.97^\circ$	$4.56 \pm 2.46^\circ$	7.7 ± 4.8
S3	$2.69 \pm 1.28^\circ$	$9.74 \pm 4.86^\circ$	$10.32 \pm 5.53^\circ$	24.8 ± 15.4
S4	$2.79 \pm 2.42^\circ$	$17.28 \pm 16.96^\circ$	$6.73 \pm 6.64^\circ$	19.8 ± 15.7
S5	$3.23 \pm 1.86^\circ$	$45.13 \pm 16.64^\circ$	$20.02 \pm 8.28^\circ$	11.8 ± 15.7
S6	$2.56 \pm 1.64^\circ$	$32.41 \pm 11.85^\circ$	$10.18 \pm 7.07^\circ$	21.0 ± 15.1
S7	$2.17 \pm 1.29^\circ$	$8.51 \pm 11.48^\circ$	$3.47 \pm 3.29^\circ$	8.1 ± 5.3
S8	$2.30 \pm 0.95^\circ$	$25.14 \pm 13.48^\circ$	$2.79 \pm 1.59^\circ$	12.2 ± 8.1
Avg.	$2.37 \pm 1.42^\circ$	$20.15 \pm 11.46^\circ$	$7.76 \pm 4.67^\circ$	13.6 ± 9.7

TABLE 4: A failure case due to large head motion.

	Proposed method	Direct estimation	Face alignment	Head motion [mm]
S9	$5.18 \pm 2.62^\circ$	$45.70 \pm 19.56^\circ$	$17.08 \pm 11.67^\circ$	32.3 ± 52.8

direct estimation performed poorly because a small misalignment of the eye region causes significant errors as explained before. On the other hand, the proposed and face alignment-based methods clearly improve the accuracy, which demonstrates that slight head motion can be well accounted for using sufficient alignment. Moreover, our method clearly outperforms the traditional face alignment-based method. By examining the alignment results, we found that even the state-of-the-art face alignment method has certain limitations in terms of accuracy, and thus, small misalignments will cause significant errors in the gaze estimation as explained in Section 3. While for our method, alignment is simultaneously optimized with the gaze estimation, which serves as a prior for achieving the optimal solution.

Note that different accuracies are achieved for different subjects in Table 3. This difference mainly depends on the magnitude of the head motion since its geometric affect on the gaze direction cannot be corrected by using the alignment method. In Table 3 we also show head movement (in translation) ranges in average and variance, which indicate the strong relationship between the gaze estimation errors and the magnitude of slight head motion ranges.

Moreover, unsatisfactory results can be seen if our assumption, *i.e.*, the head motion should be *slight*, is no longer held. One example is shown in Table 4, where the head motion range, especially its variance, is very large compared to those in Table 3. Although our method still provides the smallest error, the accuracy is degraded due to large head motion. Fortunately, to keep a relatively stable head pose is not too difficult a task commonly.

Overall, the average accuracy of 2.37° is quite satisfactory for ordinary users with slight head motion. Many of the common applications can be well supported by this proposed technique without requiring a chinrest.

TABLE 5: Fixed-head pose errors for varying resolutions.

	Proposed	Local region [16]	HOG+SVR [20]	CSLBP+GPR [21]
50%	0.60°	0.86°	0.99°	1.41°
25%	0.61°	0.84°	1.12°	1.81°
10%	0.64°	0.85°	1.36°	2.54°

TABLE 6: Gaze estimation errors for slight head motion and varying resolutions by using our alignment method.

	100%	50%	25%	10%
Subject 1	1.49 ± 0.97°	1.53 ± 1.03°	1.66 ± 1.19°	2.93 ± 1.88°
Subject 2	1.69 ± 0.98°	1.75 ± 1.05°	1.85 ± 1.18°	2.42 ± 1.16°
Subject 3	2.69 ± 1.28°	2.91 ± 1.47°	3.29 ± 2.91°	3.20 ± 1.74°
Subject 4	2.79 ± 2.42°	2.82 ± 3.09°	2.72 ± 2.52°	3.73 ± 3.23°
Subject 5	3.23 ± 1.86°	3.56 ± 1.94°	3.56 ± 1.95°	4.74 ± 3.01°
Subject 6	2.56 ± 1.64°	2.76 ± 1.55°	2.46 ± 1.17°	3.49 ± 1.91°
Subject 7	2.17 ± 1.29°	2.40 ± 1.67°	1.98 ± 1.70°	2.85 ± 1.95°
Subject 8	2.30 ± 0.95°	2.66 ± 1.27°	2.82 ± 2.03°	4.33 ± 2.28°
Average	2.37 ± 1.42°	2.55 ± 1.64°	2.54 ± 1.83°	3.46 ± 2.14°

5.3.2 Low resolution case

This section examines the efficacy of the proposed method in work with low resolution test images. In practice, low-resolution images can be captured by simply adjusting the camera configuration. However, to make a direct comparison in our experiments, we used the down-sampled versions of the full-resolution test images with scaling ratios of 50%, 20% and 10% in both width and height. In particular, the resolution of the eye region in the last case is approximately 15×10 pixels.

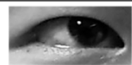



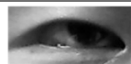
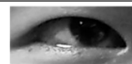
First, we conducted experiments to see how different methods perform with low resolution images under a fixed head pose. Table 5 shows the results, where gaze estimation accuracies for most methods, especially in our case, do not vary greatly against resolution changes. This demonstrates an advantage of the appearance-based methods, *i.e.*, they handle low resolution images well. Then, we examine how low resolution test images affect the gaze estimation accuracy in the cases of slight head motions. In Table 6, the errors are quite stable against resolution changes, which shows the effectiveness of our alignment method in handling low resolution images.

5.3.3 Blink detection

This section evaluates the blink detection approach described in Section 4.2.2. We specifically test the ability of the proposed method to accurately discover blinks while not being disturbed by gaze changes.

The experiments were done by using 33 training images, and a sequence of temporally successively captured test eye images covering several eye blinks. A chinrest was used to eliminate the effect of head motion for easier illustration of the experimental results. However, this is not necessary in practice because our method can handle slight head motion simultaneously within the same optimization process.

TABLE 7: Example of blinking. Six temporally successive eye images for a blink are shown. For each image, its minimized $\|\hat{\mathbf{w}}\|_1$ and gaze estimation error are listed.

Eye images (No. 1-3)						
Eye images (No. 4-6)						
Image No.	1	2	3	4	5	6
$\ \hat{\mathbf{w}}\ _1$	0.99	20.49	46.71	32.66	11.32	1.15
Estimation error	0.37°	14.36°	13.41°	14.23°	12.11°	1.15°

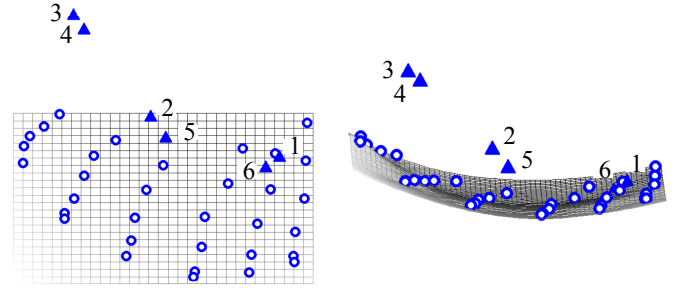


Fig. 12: **Illustration of eye features (in two views).** The eye features are projected and shown in 3-D space via PCA similarly to that in Fig. 2. The circles represent the training samples which constitute a manifold, and the triangles come from the six blinking eye images given in Table 7 using the same numbering.

Examples of the captured images during a blink can be seen in Table 7, where the blink happens after image 1 and ends in image 6. The values of $\|\hat{\mathbf{w}}\|_1$ were computed by using Eq. (8) and listed in Table 7 for all six eye images. Clearly, as declared in Section 4.2.2, the value of $\|\hat{\mathbf{w}}\|_1$ stays close to 1 for non-blink eye images, while increases significantly during blinking. It is therefore possible to discriminate between blink and non-blink images using a threshold such as $\|\hat{\mathbf{w}}\|_1 = 2$. Furthermore, gaze estimation errors are given in the table. Obviously, blinks greatly affect the estimation accuracy and that is the reason they should be detected.

In Section 4.2.2 we explain the reason for using $\|\hat{\mathbf{w}}\|_1$ in blink detection from a manifold point-of-view. This is further visualized in Fig. 12. As described in Section 2.2, normal eye image features constitute a low-dimensional manifold due to the 2-D eyeball rotation. Therefore, we project the training samples onto a 3-D space via PCA (circles in Fig. 12). The six eye images in Table 7 are also plotted (triangles with numbers). Clearly, as stated in Section 4.2.2, only test images 1 and 6, which were not captured during a blink, live in the same manifold as the training samples with $\|\hat{\mathbf{w}}\|_1 \approx 1$. On the other hand, blinking eye images 2-5 introduce extra degrees of freedom and stay far away from the manifold with $\|\hat{\mathbf{w}}\|_1 > 1$. Therefore, the criterion of $\|\hat{\mathbf{w}}\|_1 \approx 1$, which indicates whether a test image lives in the target manifold, detects outliers as blinking images.

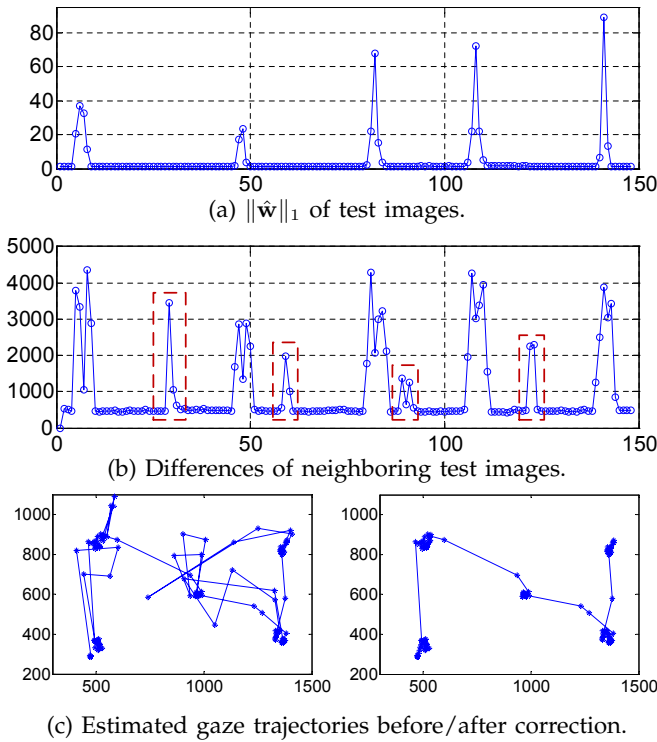


Fig. 13: **Blink detection and gaze correction.** (a) Five blinks can be correctly detected using the $\|\hat{\mathbf{w}}\|_1$ values by using our method. (b) Results of conventional methods by checking image differences. The dashed rectangles indicate the image differences caused by gaze changes that are incorrectly recognized as blinks. (c) Originally estimated gaze trajectory in screen coordinates (**left**) is corrected (**right**) using our detected blinks.

Finally, blink detection results for a captured eye image sequence are given in Fig. 13. The sequence was captured while a user was changing gaze positions between five different regions on the screen. During the user's gazing at each region, a blink was captured. Fig. 13(a) shows the detected blinks by using our method. Five blinks can be easily pointed out by checking the $\|\hat{\mathbf{w}}\|_1$ values. For comparison, Fig. 13(b) plots the root-mean-square differences of the neighboring eye images used as a cue by the conventional methods [36], [38] for the blink detection. Clearly, although blinks cause large image differences, the inverse is not always true because a gaze change also makes images look different. The dashed rectangles indicate the incorrectly detected blinks that are in fact gaze changes.

With our accurately detected blinks, the originally estimated gaze trajectory can be corrected by filtering the corresponding gaze positions. An example is shown in Fig. 13(c), where large sudden changes in the estimated gaze positions are effectively removed.

6 CONCLUSION AND FUTURE WORK

We propose a novel approach for human gaze estimation from the eye appearance. It is built upon an adaptive lin-

ear regression (ALR) method that optimally selects training samples for regression. Therefore, sparsely collected training samples are sufficient enough to achieve a high level of accuracy. Moreover, practical issues including slight head motion, image resolution variation, and eye blinking are handled under our unified ALR framework.

On the basis of the proposed approach, a gaze tracking system can be implemented that allows for easy calibration and is robust to multiple influential factors. Furthermore, the proposed method is capable of quick adaption for other similar problems.

However, limitations still exist. First, although we handle slight head motion well, in some scenarios it is still necessary to handle free head motion when eye image distortion is significant. For instance, Valenti *et al.* [3] proposed combining head pose and eye center estimation to compute gaze direction. Their method is feature-based that assumes cylinder face model and circular iris contour, and therefore it is theoretically different from our mapping-based method. However, as shown in [23], a mapping-based method like ours can collaborate with a head pose estimation technique such as that in [3] to further allow for free head motion. Therefore, it can be a good future work to bring the merit of our method to a similar head pose free method. Second, although we reduce the number of training samples, individual training is still necessary for every user. Therefore, regarding the scenario, one may choose the model-based methods if additional devices are available, or use our method for system simplicity and low resolution imaging. Overall, handling large head motion and completely removing training stage are currently out of scope in this work, but they suggest good future directions for this research.

REFERENCES

- [1] Z. Zhu and Q. Ji, "Novel eye gaze tracking techniques under natural head movement," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 12, pp. 2246–2260, 2007.
- [2] A. Villanueva and R. Cabeza, "A novel gaze estimation system with one calibration point," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 38, no. 4, pp. 1123–1138, 2008.
- [3] R. Valenti, N. Sebe, and T. Gevers, "Combining head pose and eye location information for gaze estimation," *IEEE Transactions on Image Processing*, vol. 21, pp. 802–815, 2012.
- [4] J. Wang, E. Sung, and R. Venkateswarlu, "Eye gaze estimation from a single image of one eye," in *ICCV*, 2003, pp. 136–143.
- [5] E. Guestrin and M. Eizenman, "General theory of remote gaze estimation using the pupil center and corneal reflections," *IEEE Trans. on Biomedical Engineering*, vol. 53, no. 6, pp. 1124–1133, 2006.
- [6] D. Yoo and M. Chung, "A novel non-intrusive eye gaze estimation using cross-ratio under large head motion," *CVIU*, vol. 98, no. 1, pp. 25–51, 2005.
- [7] F. L. Coutinho and C. H. Morimoto, "Improving head movement tolerance of cross-ratio based eye trackers," *International Journal of Computer Vision*, pp. 1–23, 2012.
- [8] J. Chen and Q. Ji, "Probabilistic gaze estimation without active personal calibration," in *CVPR*, 2011, pp. 609–616.
- [9] X. Brolly and J. Mulligan, "Implicit calibration of a remote gaze tracker," in *CVPRW*, 2004, p. 134.
- [10] D. Beymer and M. Flickner, "Eye gaze tracking using an active stereo head," in *CVPR*, 2003, pp. 451–458.
- [11] C. Nitschke, A. Nakazawa, and H. Takemura, "Display-camera calibration from eye reflections," in *CVPR*, 2009, pp. 1226–1233.

- [12] C. Morimoto and M. Mimica, "Eye gaze tracking techniques for interactive applications," *CVIU*, vol. 98, no. 1, pp. 4–24, 2005.
- [13] D. Hansen and Q. Ji, "In the eye of the beholder: A survey of models for eyes and gaze," *PAMI*, vol. 32, no. 3, pp. 478–500, 2010.
- [14] S. Baluja and D. Pomerleau, "Non-intrusive gaze tracking using artificial neural networks," in *NIPS*, vol. 6, 1994.
- [15] L. Q. Xu, D. Machin, and P. Sheppard, "A novel approach to real-time non-intrusive gaze finding," in *BMVC*, 1998, pp. 428–437.
- [16] K. Tan, D. Kriegman, and N. Ahuja, "Appearance-based eye gaze estimation," in *WACV*, 2002, pp. 191–195.
- [17] O. Williams, A. Blake, and R. Cipolla, "Sparse and semi-supervised visual mapping with the S³GP," in *CVPR*, 2006, pp. 230–237.
- [18] Y. Sugano, Y. Matsushita, and Y. Sato, "Appearance-based gaze estimation using visual saliency," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, p. 1, 2012.
- [19] B. Noris, K. Benmachiche, and A. Billard, "Calibration-free eye gaze direction detection with gaussian processes," in *VISAPP*, 2008, pp. 611–616.
- [20] F. Martinez, A. Carbone, and E. Pissaloux, "Gaze estimation using local features and non-linear regression," in *ICIP*, 2012, pp. 1961–1964.
- [21] K. Liang, Y. Chahir, M. Molina, C. Tijus, and F. Juen, "Appearance-based gaze tracking with spectral clustering and semi-supervised gaussian process regression," in *ETSA*, 2013, pp. 17–23.
- [22] Y. Sugano, Y. Matsushita, Y. Sato, and H. Koike, "An incremental learning method for unconstrained gaze estimation," in *ECCV*, 2008, pp. 656–667.
- [23] F. Lu, T. Okabe, Y. Sugano, and Y. Sato, "A head pose-free approach for appearance-based gaze estimation," in *BMVC*, 2011.
- [24] F. Lu, Y. Sugano, T. Okabe, and Y. Sato, "Head pose-free appearance-based gaze sensing via eye image synthesis," in *ICPR*, 2012.
- [25] —, "Inferring human gaze from appearance via adaptive linear regression," in *ICCV*, 2011, pp. 153–160.
- [26] K. Funes Mora and J. Odobez, "Gaze estimation from multimodal kinect data," in *CVPRW*, 2012, pp. 25–30.
- [27] —, "Person independent 3d gaze estimation from remote rgb-d cameras," in *ICIP*, 2013.
- [28] Y. li Tian, T. Kanade, and J. F. Cohn, "Multi-state based facial feature tracking and detection," *CMU Robotics Institute, Tech. Rep.*, 1999.
- [29] E. Amaldi and V. Kann, "On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems," *Theoretical Computer Science*, vol. 209, no. 1-2, pp. 237–260, 1998.
- [30] E. Candès and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.
- [31] D. Donoho, "For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution," *Communications on Pure and Applied Mathematics*, vol. 59, no. 6, pp. 797–829, 2006.
- [32] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *PAMI*, vol. 31, no. 2, pp. 210–227, 2008.
- [33] A. Wagner, J. Wright, A. Ganesh, Z. Zhou, and Y. Ma, "Towards a practical face recognition system: robust registration and illumination by sparse representation," in *CVPR 2009*, 2009, pp. 597–604.
- [34] X. Tan, L. Qiao, W. Gao, and J. Liu, "Robust faces manifold modeling: Most expressive Vs. most Sparse criterion," in *ICCV Workshops*, 2010, pp. 139–146.
- [35] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, "Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images," in *CVPR*, 2010, pp. 763–770.
- [36] D. Torricelli, M. Goffredo, S. Conforto, and M. Schmid, "An adaptive blink detector to initialize and update a view-based remote eye gaze tracking system in a natural scenario," *Pattern Recognition Letter*, vol. 30, no. 12, pp. 1144–1150, 2009.
- [37] A. Krolak and P. Strumillo, "Vision-based eye blink monitoring system for human-computer interfacing," in *Conference on Human System Interactions*, 2008, pp. 994–998.
- [38] T. Morris, P. Blenkhorn, and F. Zaidi, "Blink detection for real-time eye tracking," *Journal of Network and Computer Applications*, vol. 25, no. 2, pp. 129–143, 2002.
- [39] R. Heishman and Z. Duric, "Using image flow to detect eye blinks in color videos," in *IEEE Workshop on Applications of Computer Vision (WACV '07)*, 2007, pp. 52–57.
- [40] M. Lalonde, D. Byrns, L. Gagnon, N. Teasdale, and D. Laurendeau, "Real-time eye blink detection with gpu-based sift tracking," in *Fourth Canadian Conference on Computer and Robot Vision (CRV '07)*, 2007, pp. 481–487.
- [41] I. Bacivarov, M. Ionita, and P. Corcoran, "Statistical models of appearance for eye tracking and eye-blink detection and measurement," *Consumer Electronics, IEEE Transactions on*, vol. 54, no. 3, pp. 1312–1320, 2008.
- [42] H. Tan and Y.-J. Zhang, "Detecting eye blink states by tracking iris and eyelids," *Pattern Recognition Letter*, vol. 27, no. 6, pp. 667–675, 2006.
- [43] "L1 magic," <http://users.ece.gatech.edu/~justin/l1magic>.
- [44] S. Machines, "faceAPI," <http://www.seeingmachines.com>.



Feng Lu received the BS and MS degrees in Automation from Tsinghua University in 2007 and 2010, and the PhD degree in information science and technology from the University of Tokyo in 2013 respectively. He is currently a project researcher in the Institute of Industrial Science, the University of Tokyo. His research interests include human gaze analysis, photometric stereo and reflectance analysis.



Yusuke Sugano received his B.S., M.S. and Ph.D. degrees in information science and technology from the University of Tokyo in 2005, 2007 and 2010 respectively. He is currently a Project Research Associate at Institute of Industrial Science, the University of Tokyo. His research interests include computer vision and human-computer interaction.



Takahiro Okabe received the B.S. and M.S. degrees in physics, and the Ph.D. degree in information science and technology from the University of Tokyo, Japan, in 1997, 1999, and 2011 respectively. After working at the Institute of Industrial Science, the University of Tokyo, he joined Kyushu Institute of Technology, Japan, as an associate professor in 2013. His research interests include computer vision, image processing, pattern recognition, and computer graphics, in particular their physical and mathematical aspects.



Yoichi Sato received the BS degree from the University of Tokyo in 1990, and the MS and PhD degrees in robotics from the School of Computer Science, Carnegie Mellon University, in 1993 and 1997, respectively. He is a professor at the Institute of Industrial Science, the University of Tokyo. His research interests include physics-based vision, reflectance analysis, and gaze and gesture analysis. He served/is serving to conference organization and journal editorial roles including *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *International Journal of Computer Vision*, *IPSI Journal of Computer Vision and Applications*, *ECCV2012 Program Co-Chair* and *MVA2013 General Chair*.