IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 18, 2019, accepted October 11, 2019, date of publication October 23, 2019, date of current version November 5, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2949150

Training a Camera to Perform Long-Distance Eye Tracking by Another Eye-Tracker

WENYU LI[1], QINGLIN DONG[2], HAO JIA[1], SHIJIE ZHAO[3], YONGCHEN WANG[1], LI XIE[4], QIANG PAN[5], FENG DUAN[1], AND TIANMING LIU[2], (Senior Member, IEEE)

[1]Department of Automation and Intelligence, College of Artificial Intelligence, Nankai University, Tianjin 300350, China
[2]Cortical Architecture Imaging and Discovery Lab, University of Georgia, Athens, GA 30602, USA
[3]School of Automation, Northwestern Polytechnical University, Xi'an 710072, China
[4]Department of Instrument Science and Technology, Zhejiang University, Hangzhou 310027, China
[5]Xi'an iSoftStone Network Technology Company Ltd., Xi'an 710100, China

Corresponding authors: Feng Duan (duanf@nankai.edu.cn) and Tianming Liu (tliu@uga.edu)

The work of F. Duan was supported in part by the National Natural Science Foundation of China under Grant 61673224.

ABSTRACT Appearance-based gaze estimation techniques have been greatly advanced in these years. However, using a single camera for appearance-based gaze estimation has been limited to short distance in previous studies. In addition, labeling of training samples has been a time-consuming and unfriendly step in previous appearance-based gaze estimation studies. To bridge these significant gaps, this paper presents a new long-distance gaze estimation paradigm: train a camera to perform eye tracking by another eye tracker, named Learning-based Single Camera eye tracker (LSC eye-tracker). In the training stage, the LSC eye-tracker simultaneously acquired gaze data by a commercial trainer eye tracker and face appearance images by a long-distance trainee camera, based on which deep convolutional neural network (CNN) models are utilized to learn the mapping from appearance images to gazes. In the application stage, the LSC eye-tracker works alone to predict gazes based on the acquired appearance images by the single camera and the trained CNN models. Our experimental results show that the LSC eye-tracker enables both population-based eye tracking and personalized eye tracking with promising accuracy and performance.

INDEX TERMS Eye tracking, gaze estimation, human-computer interaction, machine learning.

I. INTRODUCTION

Gaze estimation has been studied over a few decades and it continues to remain as an interesting research topic [1], [2]. The main purpose of gaze estimation is to measure the viewer's gaze point on the display screen and/or particular objects. Application domains of gaze estimation include medical diagnoses and analysis [3], human computer interaction [4], [5], psychological research [6], [7], computer vision [1], product design [8], among many other areas [1], [8]. The equipment and methodology for eye tracking have also evolved for several generations [2]. The earliest generation of eye moving measurement consists of scleral contact lens, search coil, EOG and etc. [2]. In the second generation of eye tracking equipment, researchers developed photo-oculography and video-oculography for gaze estimation [2]. Then, in the third and fourth generations, analog/digital video-based methods that combined with pupil/corneal reflection were widely employed [9], [10]. These methods have also been significantly augmented by computer vision techniques and digital signal processors [2].

More recently, data-driven gaze estimation technology has developed rapidly and attracted much attention, especially camera-based appearance image gaze estimation method through computer vision [1], [11]. In the early stage, image template- and holistic-based methods were widely applied, which can alleviate the effect of varying illuminations. For instance, Zhu *et al.* [12] used Support Vector Machine (SVM) and mean shift tracking to detect the eyes in common illuminations. Samaria and Young [13] applied Hidden Markov Models (HMM) to identify faces with different facial expressions and lighting patterns. Also, machine learning methods such as subspace transformation can help to reduce the computation costs [14]. With respect to such early methods, eigenface and templates for multi-scale representation were typically used. However, these early methods detected

The associate editor coordinating the review of this manuscript and approving it for publication was Shovan Barma.

faces and eyes with relatively low accuracy and it is necessary/better to obtain other precise information for more accurate gaze estimation. Along this direction, later methods used features like wavelet [15] and Haar features [16] with conventional classifiers such as RBF and boosting algorithms like Adaboost to improve gaze estimation. However, manually-crafted image features have known limitations such as limited effectiveness and computational costs [17], [18].

With significant advancements of deep learning methodology in recent years, feature selection and end-to-end mapping has become much more powerful and promising. For example, convolutional neural network (CNN) can be trained effectively for extracting descriptive convolutional kernels as features [18], [19]. Based on this powerful methodology, the eye tracking or gaze estimation problem can be formulated as an end-to-end CNN model, i.e., from a single face image, the 2D or 3D coordinates of the human gaze point on the screen or an object can be estimated [20], [21]. In the literature, multiple studies have demonstrated the promise of this research direction. Zhang *et al.* [22] tried to estimate the gaze point in the wild based on appearance images, and then created the MPIIGaze dataset from laptop users containing more than 213K image samples. They also presented evaluations of different state-of-the-art gaze estimation methods based on the MPIIGaze dataset. However, the participant human subjects needed to concentrate on looking at the specified points, and key-pressed interactions with the laptop were required to ensure the subject's attention. As a consequence, the experimental paradigm in [22] is essentially obtained from static images, which ignored the human eye glances, continuous movements, and etc. Krafka *et al.* [23] developed a mobile App which can collect human subject's facial image and gaze point through interaction, called GazeCapture. Through training by iTracker [23], which is actually a convolutional neural network (CNN) designed for eye tracking, they obtained more data and lower error compared with previous works [23]. Zhang *et al.* [21] proposed an appearance-based method only using single full-face image by CNN. These weights are applied on the feature maps to distinguish the importance of different facial regions. The authors reported that they had achieved great improvement both on MPIIGaze and EYEDIAP dataset for 3D gaze estimation [21]. This work essentially indicated that the full face image contain the key information for gaze estimation, which inspired the work in this paper.

While these appearance-based eye tracking approaches [20]–[23] have made great advancements, however, they can be further significantly improved in several key aspects. As we all known, deep learning models such as CNNs heavily depend on the scale and quality of the training data, but collecting a large number of high-quality training samples is very costly and time consuming [24]. For instance, the MPIIGaze dataset took several months to collect 213K images from 15 participants [22], and these datasets do not necessarily contain all kinds of conditions such as different head poses and gaze directions.

Second, state-of-the-art commercial eye trackers have limited tracking distance as summarized in [25]. For instances, current popular remote eye trackers like EyeLink 1000Plus, Tobii T60XL and Tobii Pro TX300, manufactured by SR research, have recommended tracking distances less than 65 cm. In fact, long-distance eye tracking, e.g., over 100cm or even more, has a variety of important applications. For instance, an eye tracker that can monitor human's eyes in long distance, and it could be installed on the indoor ceiling or telegraph pole far away to obtain the human's gaze point, which can provide a new human-machine interface in a wider space. Also, the eye tracking system can be combined with robots for interaction in distance in order to control or acquire feedback. Thus, the related applications would no longer be confined to screen media but it is beyond current eye trackers' capability.

From our perspective, the challenges associated with current eye trackers come from several factors. First, the infrared used in commercial eye trackers based on pupil/corneal reflection scatters significantly in the process of reflection and other illuminants may also interfere in the range of long-distance measurement. As a result, those pupil/corneal reflection-based eye trackers are fundamentally limited in their tracking distance, warranting the development of novel approaches. Second, all appearance-based eye tracking methods entail and heavily rely on ''ground truth'' label information that needs the participant subjects to interact with the eye tracking system, in which process time delay and efficiency decline are inevitable. Also, those appearance-based gaze estimation methods' applications are limited to short-distance scenarios. More specifically, the eye tracking devices constrain the users to keep a close distance with the camera and attached devices, e.g., mobile phones, laptops and desktop computer screens.

To overcome the above-mentioned key limitations, this paper presents a novel long-distance gaze estimation paradigm: train a single camera to perform eye tracking by another eye tracker, named Learning-based Single Camera eye tracker (LSC eye-tracker). The paradigm is composed of two stages: training and application. In the training stage, the LSC eye-tracker acquired eye gaze data by a commercial trainer eye tracker and appearance images of faces/eyes by another long-distance trainee camera simultaneously, based on which deep convolutional neural network models are employed to learn an end-to-end mapping from appearance images to eye gazes. In the application stage, the LSC eye-tracker directly predicts eye gazes based on the acquired appearance images by the single camera and the trained CNN models.

The innovations and contributions of this LSC eye-tracker are threefold. First, we significantly relax the distance limitation for appearance-based gaze estimation, in this work, to 1.2m and 1.8m. The doubling and tripling of current state-of-the-art eye tracking distance limits would enable and facilitate many applications in the future. Second, the simultaneously acquired eye-tracking data together with the single

camera provides an easy and cost-effective way to labeling data for CNN training. In particular, using commercial remote infrared-based eye tracker in our LSC eye-tracker records the gaze point data in real-time with high frequency, which eliminates the need for user interaction and dramatically improves the efficiency of labeling data collection. Third, as a result of labeling efficiency improvement, it is possible to collect a large number of training samples. In this paper, we have collected about 2,000,000 samples from more than 16 participating subjects in 1.2m distance and 1.8m distance with our novel LSC eye-tracker paradigm. The big data collection capability enables a big-data strategy for training effective CNN models for both population-based eye tracking and personalized eye tracking with reasonably good accuracy and performance, thus it opens up new capabilities and application scenarios for long-distance eye tracking in the future.

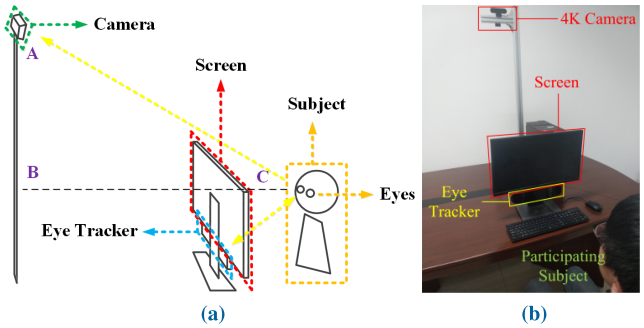


FIGURE 1. (a) Data collection system design. The trainee camera and the trainer eye tracker are installed separately in different distances. The human-eye tracker distance can satisfy the distance requirement of the commercial trainer eye tracker. The trainee camera should be installed at the top in order to capture the subject's whole face. (b) Actual experimental setup for dataset collection.

II. SYSTEM DESIGN

This section introduces our system design and articulates how to achieve long-distance gaze estimation based on appearance images acquired by a single camera. In order to achieve such a long-distance gaze estimation system, there are three important problems to be solved. First, different from the previous appearance-based short-distance gaze estimation, the camera for appearance images capture in our LSC eye-tracker needs to be placed far from users, which causes the acquired face image resolution decreasing significantly. Also, large irrelevant areas of the background in the face image need to be dealt with effectively. Second, for a learning-based data-driven system, annotating the eye gaze label is extremely important, which influences the learning-based system performance dramatically. Third, collecting a large number of samples costs the participant subjects more time, and thus the data recording efficiency needs to be considered carefully. After data collection, the data can be used to train models in different ways, and then they can be used to predict the user's gaze point on the screen via the single long-distance camera alone.

To solve the problems mentioned above, in our proposed LSC eye-tracker system, as illustrated in Fig. 1, firstly, a single camera for long-distance appearance image collection (called trainee eye-tracker) was installed on a support, which is movable and the height of the camera holder on the support can also be adjusted. Second, in order to obtain the eye gaze label with high efficiency and accuracy, we used a commercial eye-tracker (SciEyeTM aSeePro 2.0) to record the gaze point data (called trainer eye-tracker). The trainer eye-tracker estimates the gaze points on the screen based on infrared reflection with high accuracy and speed. However, there is a constraint for the trainer eye-tracker: it should be placed close to the user, typically around the display, as illustrated in Fig. 1. So we placed the trainer eye-tracker under the screen regularly, rather than next to the camera. With this arrangement, although the trainee camera for appearance image capture and trainer eye-tracker for eye gaze label collection were placed separately, as long as they work simultaneously and the data from them can be synchronized correctly. Conceptually, this is one of the major methodological innovations and contributions of this work. After the collection of the appearance images and gaze points data, CNN and fully-connected (FC) neural networks were used to train the end-to-end system, and different models and training modes were applied to test the system. Other details of the proposed LSC eye-tracker, including both of the trainee camera and trainer eye-tracker, are introduced and discussed in the Supplemental Material.

Regarding the hardware design of our LSC eye-tracker system, the trainee camera was placed at different distances away from the top of the screen. In case of collecting the facial image of the human subject, a certain height is needed to install the camera. Because the trainee camera needs to capture the whole face over the screen, the trainer eye tracker was placed under the screen. Furthermore, there should be no occlusion between the eye tracker and the eyes. In order to implement such scenario, for the 1.2m distance condition, we set the height of the camera support at 90cm (AB, as illustrated in Fig. 1a)), and the horizontal distance to be about 80cm (BC), which obtains about 120cm (AC) from the camera to the eyes. As for 1.8m condition, the height of the camera become 120cm (AB), and 130cm for BC, which obtains about 178cm (about 1.8m, as the participated subject may move slightly). In these two conditions, the obtained appearance image may be in different resolution. We resized them into same size for easily comparison. If the participant subject wears glasses, the glasses should be colorless and transparent, and the glasses frame should not obscure the path from the eyes to both the camera and the eye tracker.

During the experiments, the trainer eye tracker and the trainee camera work at the same time, and they simultaneously record the image and eye gaze point on the screen in each image frame. Each valid gaze point corresponds to an image in the collected dataset. In this way, the labels of the gaze points were obtained by the trainer eye tracker and the corresponding facial image was collected as well, which offers an easier and effective approach to obtaining the label

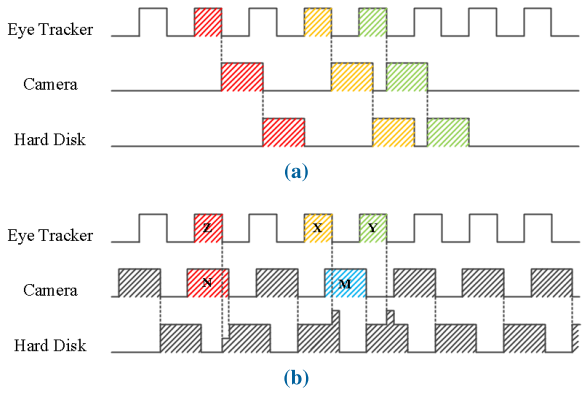


FIGURE 2. Data collection states in (a) synchronous mode and (b) asynchronous mode. The shadow within the time line indicates that the device has been activated to store the data. The eye tracker will be activated only when it has detected the subject's eyes.

for each image. As the infrared-based commercial eye tracker has high precision, this method works much faster and more effective than those using human-computer interaction-based strategies [18], [21], [23] that required viewers to stare at the given points with key-press and with long interval for each sample time.

Furthermore, as the distance between the eyes and the eye tracker is fixed to satisfy the requirement of the device, the camera is free to move and rotate disregarding the regular limitations. It is also more reasonable and accurate than using synthetic methods to obtain extra images in different angles. In order to test the influence of different distances, we conducted two sets of experiments and the distances from the trainee camera to the human eyes were set as 1.2m and 1.8m, respectively. In order to reach such a fast sample speed and maintain high precision of the screen gaze point label, we used an infrared-based commercial eye tracker as a conventional remote gaze estimation device, which can satisfy this situation perfectly. The trainer eye-tracker was first calibrated with its own software before use. Its acquisition frequency can reach 40Hz. Since we need to record the gaze fixation point of the eyes while acquiring the appearance image, and this involves the synchronization of the eye-tracker and the camera capture.

In the same loop of the recording program, we first collected 1 frame of the eye movement data, and judge the availability of the data. In fact, the eye-tracker may not have detected the eye fixation sometimes. Under the condition that the gaze point is available, the image was stored serially on the disk as well as the gaze point data. Otherwise, either the image or the gaze point data will not be stored as illustrated in Fig. 2a. In this way, each image can correspond to a gaze point label containing 4 float numbers which were the normalized coordinates of both left and right eyes on the screen.

In another way, considering the asynchronous method, while programming in different threads or processes, the sensor sampling rates can be improved. However, time stamp for both image and gaze point needs to be recorded and to be corresponded afterwards and cannot satisfy to real-time utilizations. In such a kind of fusion system, the sampling process follows event-driven theory. The data alignment can cause a mass and also reduce the accuracy of data correspondence. As illustrated in Fig. 2b, each images captured by the camera must be stored. And due to the collecting and recording speed differences between the two sensors, it is difficult to match the time stamp as demonstrated that like which gaze data (X or Y) should be matched to image M? Otherwise, it costs more to save all the data without efficiency discrimination.

While considering these cases, we use synchronization method in practice. The sampling rate can reach 2.5Hz with the resolution of 4096×2160 and about 10Hz with the resolution of 1920×1080. Although the frame rates were still a bit low, it can satisfy our application in real-time as the human eye's time resolution is lower than 10Hz. Comparing with those interaction-based with long interval label recording system, our proposed paradigm can also be used to analyze the instantaneous gaze variation. And with the speed-up of the recording hardware, the collecting frequency speed can be improved in the future.

III. PROCESSING STEPS

Since we expect to obtain the eye gaze estimation on the screen from an appearance image captured by the trainee camera, the image must contain human subject's face and eyes. In previous literature studies, by using remote eye trackers, images were captured from short distance and the eye-tracker's cameras were placed around the display screen. Therefore, the facial image typically occupied a large proportion of the whole image. In comparison, for the long-distance gaze estimation in this work, the facial image occupies a small proportion of the whole image as the trainee camera was installed substantially farther than the short distance scenarios. Furthermore, for a less constrained environment, if the human subject freely moves the body at any time, the camera may only capture the side face image. In this situation, multi-camera-based methods were investigated in the literature. For instance, multi-cameras were used to extend the detecting range in case of different head orientations [26]. Other research studies focused on adaptively fusing multiple independent cameras in order to enhance the gaze estimation accuracy [27]. Also, in the head-mounted gaze estimation system, multiple cameras were used for detecting the eyes and the head separately [9]. At this stage, our proposed LSC eye-tracker system uses a single camera as we focus on simpler and low-cost solution for now, and the multi-camera system will be explored in our future studies.

Notably, the whole image captured from a long-distance trainee camera may contain irrelevant background. In order to segment the facial area, we first performed necessary preprocessing steps to obtain the facial image. In addition, the features of the face can better reflect the orientation of the viewing angle. When the human eye is observing an object, it is not merely involving eyeball movement.

While scanning a large viewing area, the head also shifts with it. During the LSC eye-tracker data collection experiment, we asked participated subjects to watch unfamiliar videos, fix the position of the seat, and keep watching the screen. However, the participant subjects will inevitably have blinks and other behaviors in which they do not look at the screen. These cases have been a challenging problem for all kinds of eye trackers [28], [29]. For such a difficult situation, we analyze, in real-time, whether the eye-tracker has successfully detected the pupil and also discard the real-time image samples with unidentified pupils. This strategy would reduce the data sampling frequency, but it can facilitate the subsequent processing steps. These operations were realized based on the SDK provided by the commercial eye tracker. If the eye tracker does not detect the eyes, the sampled images will be discarded, which may decrease the frame rates for different participated individuals.

As used in many face image analysis applications, facial landmarks are points that sketch the outline information of the face. It can partly reflect the orientation of the head and the direction of the pupil's gaze. In order to find out whether these facial landmarks can be used to enhance the performance of the gaze estimation, we tried two publicly available tools which can extract facial landmarks for verification. For example, OpenFace [30] is an open-source facial behavior analysis toolkit, which was developed for computer vision and machine learning, interactive applications, and facial behavior analysis. It can be used for facial landmarks detection [31], head pose tracking [32], facial action unit recognition [33], and even gaze tracking [34]. The facial landmarks are extracted by a novel local detector, Convolutional Experts Network (CEN), and they also proposed an algorithm called Convolutional Experts Constrained Local Model (CE-CLM) and tested it on public datasets [31]. We used this toolkit for facial landmarks detection and compared with another commercial web interface called Face++ [35] by Megvii Technology. It provides web APIs for obtaining faces and landmarks from static images but it is not open-source.

The facial landmarks extracted by these two tools are the contour feature points of the face, including the contour features of the eyes, nose, mouth and etc. Landmark examples by OpenFace CLM-framework are illustrated in Fig. 3a. We obtained and stored the facial ROIs through both OpenFace CLM-framework and Face++. The difference between these two tools is that OpenFace CLM-framework performs background subtraction on the original image according to the outer contour of the face and performs spatial rotation transformation according to the predicted head orientation. In comparison, Face++ only selects the face part of the original image through the square box and it does not perform other transformation operations, as illustrated in Fig. 3b, 3c. Additional details of the facial landmarks and other information obtained by these two tools are shown in Table 1.

In order to obtain high-quality training samples, we manually screened the collected samples after the facial image area was obtained, and filtered out the images that identified

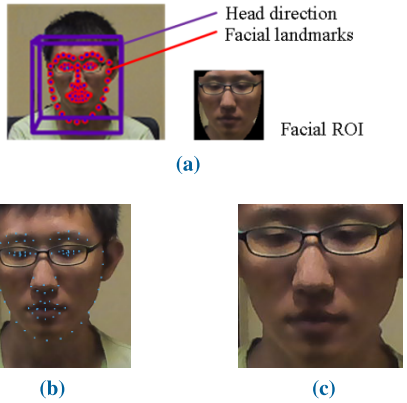


FIGURE 3. (a) OpenFace CLM-framework extracted facial ROI and facial landmark features. The facial landmarks include the facial edge and the outlines of the facial features. Head direction can be also evaluated. It can also process background elimination. (b) Landmarks extracted by Face++ (c) Facial image extracted by Face++.

TABLE 1. Comparison between OpenFace and Face++.

Item	OpenFace	Face++
Basic method	HOG	Unavailable
RGB image	Yes	Yes
Number of the landmarks	65	83/106
Background subtraction	Yes	No
Head direction estimation	Yes	Yes
Gaze direction estimation	Yes	Yes
Facial bounding box	Yes	Yes
Facial image projection	3D affine	3D affine
Resolution supported	No limitation	$48^2 - 4096^2$
Interface	Open source	Web API

the wrong ROIs containing no face. Our results show that the OpenFace CLM-framework has a higher error rate than Face++. In addition, among the facial landmarks obtained by these two tools, there are some parameters related to the screen size. Although we use the same screen in our entire data collection procedure for different distances, the normalized processing would help us to have more samples of different sizes later. And all these tests mentioned above are processed on the dataset with the image resolution of 4096×2160 pixels.

However, these two face analysis tools are very computationally expensive and time-consuming. Although the OpenFace CLM-framework can run locally, it costs about 3s for an individual high-resolution image. And as for Face++ web API, the time is consumed on data transmission through the network. Each image costs about 12s, without considering the possible failure of network itself. Notably, YOLO [36] was also exploited in this work, which is a state-of-the-art real-time object detection system that has a very fast (e.g., 50 fps) and good performance. The accuracy of the prediction by YOLO also depends on the training inputs, and more samples are needed especially for particular application like eye tracking. However, the segmentation of the face by YOLO is not as good as the two tools mentioned above, and YOLO cannot extract the facial landmarks neither. Thus, in this

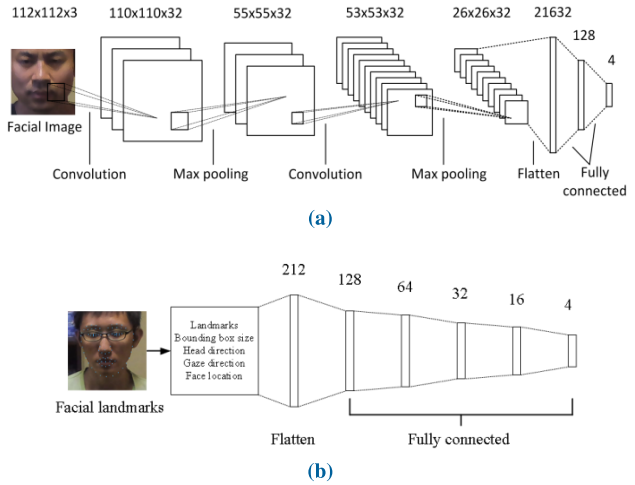


FIGURE 4. (a) Deep learning model uses facial image only (type 1). Preprocessing tools can obtain the facial image. Two convolutional and pooling layers were used and then followed by FC layers. The outputs are the normalized gaze point coordinates. (b) Deep learning model uses facial landmarks and other features only (type 2). The model uses FC structure through the whole network. The outputs are the normalized gaze point coordinates, which are the same as in (a).

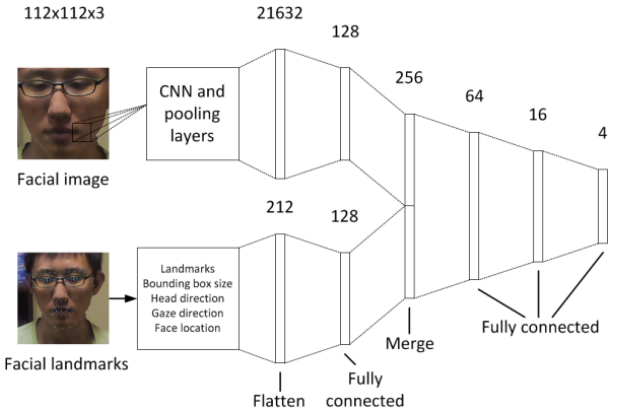


FIGURE 5. Combination model uses both the facial image and landmark features (type 3). The upper part of the model is similar to the facial image model. As for facial landmarks part, landmarks, bounding box size, head direction, gaze direction and face location extracted by the preprocessing tools are all used as the features for training.

paper, we first used the OpenFace CLM-framework and Face++ for preprocessing the collected face images, as these two tools can automatically extract the facial images with finer boundary and facial landmarks. It should be pointed out that they were only used for data collection and model training, without considering time consumptions. Speed-up and optimization will be explored in our future studies.

IV. DEEP LEARNING MODELS

The state-of-the-art deep learning works such as [18], [21]–[23] all used multi-CNN and FC layers. However, different network structures are utilized in terms of different data types. References [18] and [22] only used eye images for gaze estimation, in which multi-CNN and FC layers are applied. Reference [21] used facial images instead, and applied spatial weights for improving. Furthermore, reference [23] used both eye image, facial image and face mask, in which 4 CNN pipe lines were used for processing and feature extraction, then combined through FC layers, which inspired us to apply multi-feature to improve the performance. Hence, we designed three types of deep network models to test the effectiveness and efficiency of facial image and landmarks data, as well as their combination, as illustrated in Fig. 4a, 4b and Fig. 5.

All these three models use the normalized screen coordinates obtained by the trainer eye tracker as the assumed ''ground truth''. For the first type, only facial images were used for training (type 1). Then we used landmark features individually (type 2) and their combination was used to train the model together (type 3). These three models were tested to see whether/how the landmark features contain useful information for eye gaze estimation and if yes, how much contribution they make.

For the type 1 model, we assume that the facial images can reflect the position of the human eyes gaze point when the human body does not move with head while watching the screen. Regarding the CNN models, the activation function of each CNN layer adopts Relu, and after each max pooling, a layer of dropout is added to prevent over-fitting. After the second max pooling layer, all nodes are expanded into a single row, and then the FC neural network is used for fitting. The model is illustrated in Fig. 4a.

For the type 2 model, we only used the facial landmarks and other features as the feature vectors for training obtained by the OpenFace CLM-framework and Face++, respectively. The model adopted is an FC network. The numbers of nodes in each layer are 128, 64, 32, 16 and 4, respectively. The activation function for each CNN layer is Relu, and dropout is also used to prevent over-fitting for each layer. Among these three types of models, the ground truth for each model is the same. Four values for each sample contain the normalized binocular screen fixation x and y coordinates.

For the type 3 model, by using such an FC layer, if the training and test results were close to those using facial image only or their combination, we believe that the landmarks include most of the information of human gaze point and it would be more helpful for future related model design. However, if the results were not good, we believe that there is some information more important than the landmarks within the facial image. As for the combination of them all, those landmarks were indeed extracted from the image, and we consider it as an enhancement for the facial image, which also tests the importance of those landmarks.

The combination model merges the two networks mentioned above, gradually increases the network depth, and merges after completely forming a single row and finally adopts an FC network for training, as illustrated in Fig. 5. In this way, we aim to emphasize the importance of these facial landmarks which were extracted as the meaningful contour of the face. These feature points include the bounding

box size and other relatively coordinates data can reflect part of the position information of the face in the original whole image. These additional inputs are not like those studies adding extra eye image as input. The facial landmarks and other features are not image information and may be more useful in other applications (e.g., facial expression recognition).

In summary, the input source of the CNN model is the preprocessed facial ROI by OpenFace CLM-framework or Face++. The structure of the CNN and pooling layers are the same as those in the first model which was trained with facial ROI individually. It consists of two layers of convolution and max pooling, all of which have been activated by Relu function and dropout has been used to prevent from over fitting. The node number of the flatten layer is 21,632 and then was reduced into 128 for merging with the landmark features.

The facial landmarks and other features in the model 3 include landmarks, bounding box size, head direction vector, gaze direction vector and face location coordinates. In Face++, the total number of input nodes is 212, and then was reduced into 128 nodes by FC dense neural network. Since the inputs from both facial image and landmark features are transformed into 128 nodes, we merge them together by connecting the two 128 dimensional vector into a 256 dimensional vector. Then FC layers are used for follow-up processing steps. The numbers of nodes in each dense layer are 64, 16 and 4, respectively. The ''ground truth'' used is the same as the two types of models mentioned above.

These models were tested on a small dataset with a high resolution of 4096 × 2160 pixels, including more than 100K samples of 6 subjects with both 1.2m and 1.8m distances setup. We also used a larger dataset collected under same conditions to further verify our proposed models. Only the best model (using facial image only) among those three tested on the small dataset was tested on this larger dataset. To gain the speed-up for real-time utilization, we used YOLOv3 for facial ROI image extraction as a preprocessing step. The facial images were obtained via training the YOLOv3 deep learning model. As for gaze estimation, we followed the type 1 model. Except the input source, all the hyper-parameters are all the same for these three types of CNN models.

V. EXPERIMENT SETUPS

We designed and conducted several comparative experiments. For the capturing distance of facial images, we tried the distance from the camera to the head with 1.2m and 1.8m, respectively, and each experimental data was collected and processed separately. Under different distance conditions, the position of the trainer eye tracker is the same: just below the monitor and about 60cm from the participant subject's face, as illustrated in Fig. 1b.

In addition, we used two tools to obtain facial ROIs, landmarks and other features: OpenFace CLM-framework and Face++. The OpenFace CLM-framework can obtain facial ROIs and 234 dimensional landmarks and other features. Face++ can intercept square faces in the original image and obtain 211 dimensional landmarks features. For different situations, this paper uses three different models to train and compare the performance. After this process, the best model was selected for a larger dataset test. Details of the facial image, individual model, landmarks model and combination models are given in the Supplemental Material.

For all training data, we used 10-folds cross-validation to test the model's fitting ability and the feasibility of the method. And all the models are trained based on each subject's data individually. Then we evaluated the mean errors of the results. Six participants have conducted the experiments and each one was asked to watch one video that he or she had never watched before. They were asked to participate in both 1.2m and 1.8m distance experiments. And each experiment lasted about 2.5 hours and collected about 10K image samples.

In the larger dataset experiments, we used lower resolution (1920 × 1080) to capture the images in order to obtain more samples and reduce the cost and time consumption of the subjects. In total, more than 16 subjects attended the experiment and both 1.2m and 1.8m distances were tested. We have collected around 50K samples of each subject. Each collection process lasted about 1.5 hours. After that, we used YOLOv3 to extract the facial ROIs and then did the training processes with the best model that we had obtained from the small dataset mentioned above.

Due to the loss errors declining curves, we found that the convergence rate gradually slowed down and after around 15 epochs, the loss errors did not decrease much. So we reduced the training epochs from 30 to 20. Another parameter that we changed was the facial image size. We resized the facial image extracted by YOLOv3 to 200 × 200 resolutions. We believe this can improve the performance of our model and this size is also more reasonable for comparing with other works. Other parameters of the large dataset training process were all the same, and details are provided in the Supplemental Material.

The image recording resolution and model type of the large dataset were chosen according to the experimental performance on the small dataset. On these two datasets, both 1.2m and 1.8m distance experiments were processed for comparison, and in-depth study on population-based experiment was tested on the large dataset. In order to speed up the preprocessing process and due to the poor performance on feature-based model, YOLOv3 was used to obtain the facial images instead of OpenFace CLM-framework and Face++.

VI. EXPERIMENTAL RESULTS

In the experiments, cross-validation was used to test the accuracy and performance of each experiment. Different distances, preprocessing tools and feature models were tested on the small dataset. After obtaining the results on the small dataset, an in-depth study was performed on the large dataset, in which a significantly larger number of samples were processed with faster preprocessing tool. In Table 2,

TABLE 2. Experiment design and comparison between different datasets.

Item	Small dataset	Large dataset
	(6 subjects, 10K samples, 2.5 hours each, 4096 × 2160)	(16 / 18 subjects, 50K samples, 1.5 hours each, 1920 × 1080)
Distance	1.2m / 1.8m	1.2m / 1.8m
Training mode	Individual	Individual / Population
Preprocessing tool	OpenFace CLM-framework / Face++	YOLOv3
Model type	Image only (type 1) / features only (type 2) / combination (type 3)	Image only (type 1)

experimental results and comparisons on both small dataset and large dataset are briefly summarized, respectively, which will be further detailed in the following subsections.

A. EXPERIMENTAL RESULTS ON THE SMALL DATASET

For all types of experimental comparisons with the small dataset, we used the 10-folds cross-validation to test the average error. First, we compare the effectiveness of our method with different distances. OpenFace CLM-framework and Face++ were applied to extract facial image and landmarks, as well as other features. The OpenFace CLM-framework can extract the affine-transformed face with the contoured background and the background removed, as shown in Fig. 3a. The faces extracted by Face++ have no background as shown in Fig. 3b, 3c.

Second, the differences among various types of feature models are also compared. The result of simply using facial images is much better than that of feature-based training. This result supports the assumption that appearance-based gaze estimation can use the image data to predict the gaze point directly, which uses region information and considers more than points combinations. In contrast, feature-based gaze estimation uses features which are also extracted from the image. Hence, the feature-based method estimates the gaze point indirectly and may ignore other meaningful information. Other details are illustrated in the Supplemental Materials.

As for the feature combination model's training results, the combination of both the facial image and landmark features is slightly worse than using the facial images alone. We believe it is due to that some of the feature points have no influence on the eye gaze estimation. Otherwise, according to the previous analysis, feature points such as landmarks do not adequately characterize the attributes of the eye gaze point. This phenomenon is consistent for the results of the facial image and feature extraction algorithms with different distances.

However, feature-based gaze estimation is still an efficient and person-independent method. It does not need sample collection and training in advance. If the meaningful features can be extracted effectively and quickly, it would be better than using facial image and data-driven approaches. However, how to extract such meaningful features is the key problem for

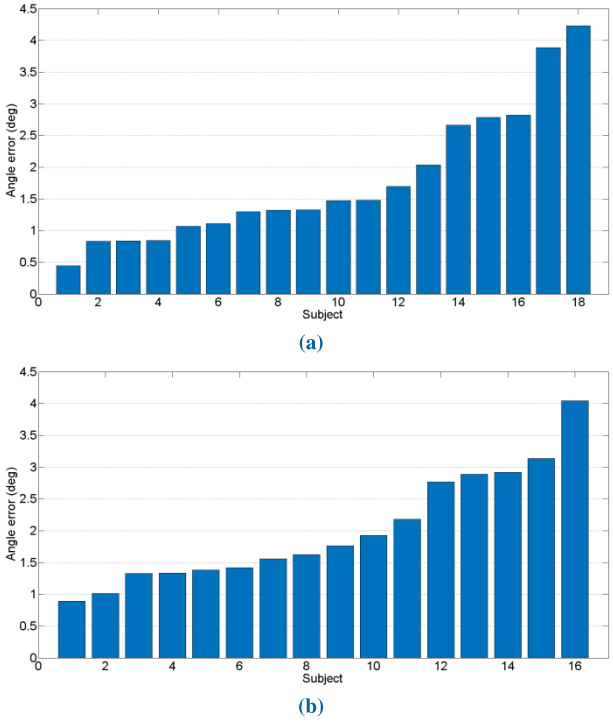


FIGURE 6. Results of different subjects were sorted. (a) Angle error with 1.2m distance condition (b) Angle error with 1.8m distance condition.

feature-based gaze estimation. More features and structural information should be explored in the future, and we plan to seek such kinds of features through the experience learned from the results of appearance-based method in this work.

Third, while comparing the two preprocessing tools, OpenFace CLM-framework and Face++, Face++ performed better than OpenFace CLM-framework with models based on facial images and feature combination. This might be due to the accuracy of the facial image extraction. As shown in Fig. 3a, the facial bounding contours by OpenFace CLM-framework are not highly accurate compared with facial landmarks in Fig. 3b by Face++, though Face++ does not perform background subtraction on facial ROI. From this point of view, accurate face areas and features are crucial for eye gaze estimation.

B. EXPERIMENTAL RESULTS ON THE LARGE DATASET

In order to further verify our strategy and the system performance, a 1920 × 1080 resolution sample dataset was collected and was first processed with the YOLOv3 tool [36] to extract the facial images. Each subject has approximately 50K samples (examples are shown in the Supplemental Material). However, due to data collection failure and individuals' characteristics, the trainer eye tracker cannot detect everyone's eyes correctly (the sampling rate was low which was always lower than 3Hz that we cannot collect enough samples of those participant subjects). Only 18 subjects with 1.2m distance and 16 subjects with 1.8 m distance were analyzed in this paper.

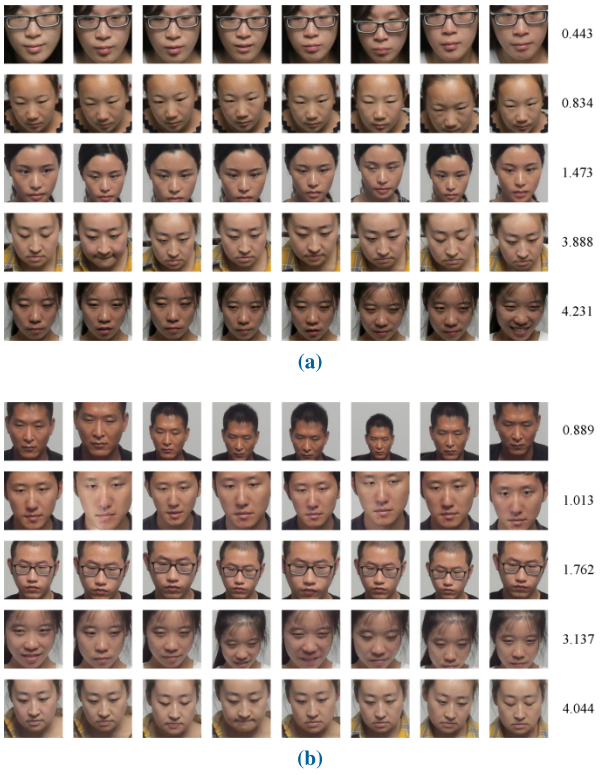


FIGURE 7. (a) 1.2m distance samples with facial ROI extracted (b) 1.8m distance samples with facial ROI extracted. In each sub-figure, each row demonstrated samples of one subject. The error of each row from top to bottom is corresponded to the smallest, the second smallest, medium, the second largest, and the largest noted on the right according to the experimental results, as illustrated in Fig. 6.

The 5-folds cross-validation mean error results are illustrated in Fig. 6. The results of all of the subjects were sorted. The mean error in 1.2m distance condition is about 1.79 degrees, and the error is 2.01 degrees for 1.8m distance condition. Fig. 7 illustrated several samples of subjects in both 1.2m distance and 1.8m distance conditions. From top to bottom, the rows demonstrated the facial ROIs of the subjects with the most accurate result, the second most accurate, medium level, the second most inaccurate, and the most inaccurate, respectively. We believe that glasses frame may shelter eyes and cause errors, however, there are few participating subjects wearing glasses (examples of participating subjects with eyes sheltered by glasses are shown in Supplemental Material) and in-depth study needs to be considered. When the trainer eye tracker cannot detect the subjects' eyes accurately, the training samples may negatively contribute to the model training. Also, the samples of subject with closed eyes also matters remarkably (examples of facial image with closed eyes are shown in the Supplemental Material). These failures need to be omitted in order to obtain usable and meaningful training samples. Notably, this type of failure could be potentially avoided in the future, by asking the participant subject to keep certain state and ensure that their glasses (if any) should not shelter eyes from the sensors.

In spite of imperfect training samples, comparing with previous studies in short distance, our results (1.79 degrees in 1.2m and 2.01 degrees in 1.8m) are considered promising according to Table 3. Notably, our method focuses on training each subject data individually, which also means person-dependent. In fact, this approach can be used for individual gaze estimation (personalized model). Once the model is trained for a subject, like a radiologist, a surgeon, or a student in professional studies, it can help the user to process remote-operation, avoid direct contact and free the user from the screen. The gaze movement differences between individuals may also be regarded as a biological feature which can be used in security scenario. In addition to these situations where personalized gaze estimation models are important, we believe it is also useful to obtain a generalization model for particular users without prior training samples collected.

TABLE 3. Distance errors comparison with other methods.

Method	Error	Description
Center	7.54	Simple baseline
Ours	**5.39**	**1.8m distance, Multi-layer CNN**
TurkerGaze [37]	4.77	Pixel features + SVR
Ours	**4.58**	**1.2m distance, Multi-layer CNN**
Zhang [21]	4.20	Spatial weights CNN
MPIIGaze [18]	3.63	CNN + head pose
TabletGaze [38]	3.17	Random forest + mHOG
iTracker [23]	2.58	Fc1 of iTracker + SVR

Second, in addition to training the personalized model individually, in order to compare with the other studies, we also evaluated population-based (person-independent) models to verify our methods. In this case, we extracted 2K samples from each subject and then formed an integrated and aggregated dataset. Experiments with different distances were conducted respectively. The same model we used in individual facial image training was applied. The comparisons with other studies are illustrated in Table 3.

Those results in Table 3 are based on the reports in literature [23], and we did not realize these methods to compare with our algorithm due to various constraints, such as the experimental environment, methods, and system design. Instead, an intuitive comparison is given in Table 3. In order to compare with those results, we also transformed our results into distance errors in cm. While transforming into sight view angle error, the 1.2m distance result is 4.37 degrees, and it is 5.13 degrees for 1.8m. Notably, we can see that our models can achieve relatively good results both in 1.2m and 1.8m while comparing with these state-of-the-art methods.

Third, among all the experiments and comparison on the large dataset, the results in both 1.2m and 1.8m are given. Comparing different distances in the experiments, the performances in 1.2m were all better than that in 1.8m. It is easy to explain that the trainee camera can capture clearer image in short distance. Facial and eye image should be much clearer and accurate, which may contribute to the better results. However, different from short-distance (about 0.6m) condition, facial image extraction is not necessary, which is fairly important in our study with long-distance condition. Also, other extensible application like remote interaction and

multi-person gaze estimation cannot be realized in short distance, which we would like to investigate in the future.

VII. CONCLUSION AND DISCUSSION

Gaze estimation, as a novel human-computer interface, has been studied and developed extensively in recent years. However, considering wider applications such as outdoor environment and real-time interaction, problems including distance limitation, discontinuous sample collection and complex interaction procedures in training process still remain to be solved. In this paper, we proposed a novel long-distance gaze estimation paradigm called LSC eye-tracker, which used a single trainee camera to acquire appearance images from long distance and a trainer commercial eye tracker for gaze data collection simultaneously. During the training stage, deep CNN models are employed to learn an end-to-end mapping from appearance images to eye gazes. In the application stage, the LSC eye-tracker predicts eye gazes based on the acquired appearance images from the single trainee camera and the trained CNN models.

The perceived innovations of this study are threefold. First, the LSC eye-tracker breaks the distance limit by placing the single trainee camera and trainer eye tracker in different distances. Both 1.2m (twice of the typical remote eye-tracking condition) and 1.8m distance (three times of the typical remote eye-tracking condition) were tested of each participating subject. Second, the simultaneous data collection strategy eliminates the need for user interaction and dramatically improves the labeled training data collection efficiency. Issues of synchronizing the appearance images to the eye gaze data were also studied in order to achieve a faster sample collection strategy. Third, the above second innovation contributes to collecting a larger number of training samples, which benefits significantly to the model training with a big-data strategy.

In the model training process, three types of feature models were tested, including using the facial image individually, using the facial landmark features individually and using their combination. We tried all of them on a small dataset with two different preprocessing tools (OpenFace CLM-framework and Face++) for facial images and landmark features extraction. The best model was using the facial image individually, and we also tested it on a larger dataset with both personalize and population-based validation. In order to speed up the preprocessing procedure, we used YOLOv3 for facial images extraction with the best model and lower image resolution. Considering multiple application scenarios, in the personalized experiments, we achieved the mean angle errors about 1.79 degrees in 1.2m distance and 2.01 degrees in 1.8m distance. In the population-based experiments, we achieved the angle errors about 4.58 degrees in 1.2m distance and 5.39 degrees in 1.8m distance. We believe these accuracies and results are state-of-the-art.

However, there are still limitations in our system that could be improved in the future. As the current LSC eye-tracker system focused on the long-distance gaze estimation and fast data collection, the experiments settings required subjects to keep static which has limitations in unconstrained real-world settings [24]. Therefore, handling head pose and eye location for gaze estimation should be enhanced for the problem of non-frontal faces [23], [39], and the addition of eye image synthesis might partly solve the head pose-free problem [40]. Furthermore, data normalization for mapping the input images and gaze labels to a normalized space [24] seems possible and useful to solve such problems. Also, more advanced CNN structures and models could be explored for such end-to-end mapping from appearance images to eye gazes.

In the future, by applying our proposed LSC eye-tracker system, applications such as controlling the indoor household appliances remotely with the development of IOT could be possible. Also, electronic entertainments like somatic games may obtain a novel feedback for human-computer interactions. Furthermore, cameras that already installed outside could extend their ability as well based on our proposed strategy. We believe that multiple applications in both scientific research and daily utilization could be enabled and advanced through our LSC eye-tracker paradigm with further improvements in the future.

ACKNOWLEDGMENT

The authors would like to thank the volunteers for their participation in the data acquisition. The authors would like to thank Jugal Kamlesh Panchal for his help in initially setting up the eye tracking system. The authors would also like to thank Yun Zhang for sharing her commercial eye tracker SciEyeTM aSeePro 2.0.

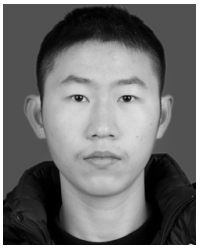
REFERENCES

[1] P. Wang and Q. Ji, "Learning discriminant features for multi-view face and eye detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2005, pp. 272–279.
[2] D. T. Andrew, *Eye Tracking Methodology: Theory and Practice*, vol. 24. London, U.K.: Springer, 2017.
[3] P. S. Holzman, L. R. Proctor, D. L. Levy, N. J. Yasillo, H. Y. Meltzer, and S. W. Hurt, "Eye-tracking dysfunctions in schizophrenic patients and their relatives," *Arch Gen Psychiatry*, vol. 31, no. 2, pp. 143–151, Aug. 1974.
[4] P. Lanillos, J. F. Ferreira, and J. Dias, "A Bayesian hierarchy for robust gaze estimation in human–robot interaction," *Int. J. Approx. Reasoning*, vol. 87, pp. 1–22, Aug. 2017.
[5] T. Strandvall, "Eye tracking in human-computer interaction and usability research," in *Human-Computer Interaction* (Lecture Notes in Computer Science). Springer-Verlag, 2009, pp. 936–937. doi: 10.1007/978-3-642-03658-3_119.
[6] D. Duan, L. Tian, J. Cui, L. Wang, H. Zha, and H. Aghajan, "Gaze estimation in children's peer-play scenarios," in *Proc. IAPR Asian Conf. Pattern Recognit. (ACPR)*, Nov. 2013, pp. 754–760.
[7] W. Cui, J. Cui, and H. Zha, "Specialized gaze estimation for children by convolutional neural network and domain adaptation," in *Proc. Int. Conf. Image Process. (ICIP)*, pp. 3305–3309, Sep. 2018, vol. 1, no. 1.
[8] G. T. Papadopoulos, K. C. Apostolakis, and P. Daras, "Gaze-based relevance feedback for realizing region-based image retrieval," *IEEE Trans. Multimedia*, vol. 16, no. 2, pp. 440–454, Feb. 2014.
[9] M. Tonsen, J. Steil, Y. Sugano, and A. Bulling, "InvisibleEye: Mobile eye tracking using multiple low-resolution cameras and learning-based gaze estimation," *ACM Interact, Mob. Wearable Ubiquitous Technol.*, vol. 1, no. 3, p. 106, Sep. 2017.
[10] K. A. F. Mora, F. Monay, and J. M. Odobez, "EYEDIAP: A database for the development and evaluation of gaze estimation algorithms from RGB and RGB-D cameras," in *Proc. Eye Track. Res. Appl. Symp.*, vol. 25, Mar. 2014, pp. 255–258.

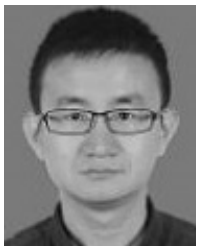
[11] F. Lu, Y. Sugano, T. Okabe, and Y. Sato, "Adaptive linear regression for appearance-based gaze estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 2033–2046, Oct. 2014.
[12] Z. Zhu, K. Fujimura, and Q. Ji, "Real-time eye detection and tracking under various light conditions," in *Proc. Eye Track. Res. Appl. Symp.*, Mar. 2002, pp. 139–144.
[13] F. Samaria and S. Young, "HMM-based architecture for face identification," *Image Vis. Comput.*, vol. 12, no. 8, pp. 537–543, Oct. 1994.
[14] D. W. Hansen and Q. Ji, "In the eye of the beholder: A survey of models for eyes and gaze," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 3, pp. 478–500, Mar. 2010.
[15] J. Huang and H. Wechsler, "Eye location using genetic algorithms," in *Proc. 2nd Int. Conf. Audio Video Biometric Person Authentication*, Mar. 1999.
[16] P. Viola and M. J. Jones, "Robust real-time face detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2. 2001, p. 747.
[17] P. Wang, M. B. Green, Q. Ji, and J. Wayman, "Automatic eye detection and its validation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 3, Sep. 2005, p. 164.
[18] K.-H. Tan, D. J. Kriegman, and N. Ahuja, "Appearance-based eye gaze estimation," in *Proc. IEEE Workshop Appl. Comput. Vis.*, Dec. 2002, pp. 191–195, vol. 1, no. 1.
[19] W. Sewell and O. Komogortsev, "Real-time eye gaze tracking with an unmodified commodity webcam employing a neural network," in *Proc. Conf. Hum. Fact. Comput. Syst. Proc.*, Apr. 2010, pp. 3739–3744.
[20] J. Chen and Q. Ji, "3D gaze estimation with a single camera without IR illumination," in *Proc. 19th Int. Conf. Pattern Recognit.*, Dec. 2008, pp. 1–4.
[21] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "It's written all over your face: Full-face appearance-based gaze estimation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jul. 2017, pp. 2299–2308, vol. 1, no. 1.
[22] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "Appearance-based gaze estimation in the wild," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4511–4520, vol. 1, no. 1.
[23] K. Krafka, A. Khosla, P. Kellnhofer, H. S. Bhandarkar, W. Matusik, and A. Torralba, "Eye tracking for everyone," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2176–2184, vol. 1, no. 1.
[24] X. Zhang, Y. Sugano, and A. Bulling, "Revisiting data normalization for appearance-based gaze estimation," in *Proc. Eye Track. Res. Appl. Symp. (ETRA)*, Jun. 2018, p. 12, vol. 1, no. 1.
[25] D. C. Niehorster, T. H. W. Cornelissen, K. Holmqvist, I. T. C. Hooge, and R. S. Hessels, "What to expect from your remote eye-tracker when participants are unrestrained," *Behav. Res. Methods*, vol. 50, no. 1, pp. 213–227, Feb. 2018.
[26] A. Utsumi, K. Okamoto, N. Hagita, and K. Takahashi, "Gaze tracking in wide area using multiple camera observations," in *Proc. Eye Track. Res. Appl. Symp.*, Mar. 2012, pp. 273–276.
[27] N. M. Arar, H. Gao, and J.-P. Thiran, "Robust gaze estimation based on adaptive fusion of multiple cameras," in *Proc. 11th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit. (FG)*, May 2015, pp. 1–7.
[28] J. Amudha, S. R. Reddy, and Y. S. Reddy, "Blink analysis using eye gaze tracker," *Adv. Intell. Sys. Comput.*, vol. 530, no. 1, pp. 237–244, Sep. 2016.
[29] F. Bernard, C. E. Deuter, P. Gemmar, and H. Schachinger, "Eyelid contour detection and tracking for startle research related eye-blink measurements from high-speed video records," *Comput. Methods Prog. Biomed.*, vol. 112, no. 1, pp. 22–37, Oct. 2013.
[30] T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L.-P. Morency, "OpenFace 2.0: Facial behavior analysis toolkit," in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2018, pp. 59–66, vol. 1, no. 1.
[31] A. Zadeh, Y. C. Lim, T. Baltrusaitis, and L.-P. Morency, "Convolutional experts constrained local model for 3D facial landmark detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2519–2528, vol. 1, no. 1.
[32] T. Baltrusaitis, P. Robinson, and L.-P. Morency, "Constrained local neural fields for robust facial landmark detection in the wild," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 1, Jun. 2013, pp. 354–361.
[33] T. Baltrušaitis, M. Mahmoud, and P. Robinson, "Cross-dataset learning and person-specific normalisation for automatic Action Unit detection," in *Proc. IEEE Int. Conf. Workshops Autom. Face Gesture Recognit.*, vol. 6, May 2015, pp. 1–6.
[34] E. Wood, T. Baltruaitis, X. Zhang, Y. Sugano, P. Robinson, and A. Bulling, "Rendering of eyes for eye-shape registration and gaze estimation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 3756–3764. vol. 1, no. 1.
[35] *Face++, Megvii*. Accessed: Jun. 4, 2018. [Online]. Available: https://www.faceplusplus.com/
[36] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," Apr. 2018, *arXiv:1804.02767*. [Online]. Available: https://arxiv.org/abs/1804.02767
[37] P. Xu, K. A. Ehinger, Y. Zhang, A. Finkelstein, S. R. Kulkarni, and J. Xiao, "Turkergaze: Crowdsourcing saliency with webcam based eye tracking," Apr. 2015, *arXiv:1504.06755*. [Online]. Available: https://arxiv.org/abs/1504.06755
[38] Q. Huang, A. Veeraraghavan, and A. Sabharwal, "TabletGaze: A dataset and baseline algorithms for unconstrained appearance-based gaze estimation in mobile tables," Aug. 2015, *arXiv:1508.01244*. [Online]. Available: https://arxiv.org/abs/1508.01244
[39] R. Valenti, N. Sebe, and T. Gevers, "Combining head pose and eye location information for gaze estimation," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 802–815, Feb. 2012.
[40] F. Lu, Y. Sugano, T. Okabe, and Y. Sato, "Head pose-free appearance-based gaze sensing via eye image synthesis," in *Proc. Int. Conf. Pattern Recognit.*, Nov. 2012, pp. 1008–1011.

WENYU LI received the B.S. degree in intelligence science and technology from the College of Information Technical Science, in 2013, and the Ph.D. degree in control science and engineering from the College of Artificial Intelligence, Nankai University, Tianjin, China, in 2019. He is currently applying for postdoctoral with the School of Vehicle and Mobility, Tsinghua University. His current research interests include gaze estimation and human-vehicle collaborative driving.

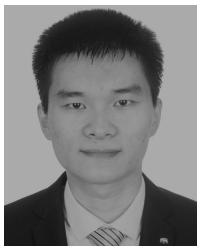
QINGLIN DONG received the Ph.D. degree in computer science from UGA, advised by Dr. Tianming Liu. He has made great contributions in the thesis of unsupervised deep learning on human brain function networks. His research interests include generative models on connectomes and discriminative models on brain decoding.



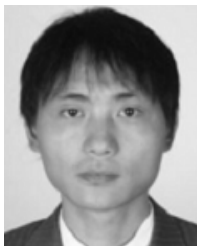
HAO JIA received the B.S. degree from the College of Computer and Control Engineering, Nankai University, China, in 2018, where he is currently pursuing the M.S. degree with the College of Artificial Intelligence. His research interests include brain function simulation and spike neural networks.



SHIJIE ZHAO received the Ph.D. degree from the School of Automation, Northwestern Polytechnical University, Xi'an, Shaanxi, China, where he is currently an Assistant Researcher in natural science. His research interests include biomedical image processing, pattern recognition, and deep learning.



YONGCHEN WANG received the B.S. degree from the College of Computer and Control Engineering, Nankai University, China, in 2016, where he is currently pursuing the M.S. degree with the College of Artificial Intelligence. His research interests include gaze estimation and computer vision.



LI XIE is currently an Associate Professor with the Department of Instrument Science and Technology, Zhejiang University, Hangzhou, Zhejiang, China. His research interests include embedded network multimedia system with image processing and short-range wireless communication technology, navigation, and control based on multi-sensor information fusion.

QIANG PAN is the Mobile Terminal Software Architect of iSoftStone Information Technology (Group) Company Ltd. His research interests include mobile communication, computer graphics, and human–computer interaction.

FENG DUAN received the M.S. and Ph.D. degrees in precision engineering from the University of Tokyo, Tokyo, Japan. He is currently a Professor with the College of Artificial Intelligence, Nankai University, Tianjin. His current research interests include cellular manufacturing, human skill, and image processing.

TIANMING LIU is currently a Distinguished Research Professor of computer science with the University of Georgia. His research interests include brain imaging and mapping. He has published over 280 peer-reviewed articles in this area. Dr. Liu was a recipient of the NIH Career Award and the NSF CAREER award, both in the area of brain mapping.