Machine Vision and Applications (2017) 28:445–461
DOI 10.1007/s00138-017-0852-4

ORIGINAL PAPER

TabletGaze: dataset and analysis for unconstrained appearance-based gaze estimation in mobile tablets

Qiong Huang[1] · **Ashok Veeraraghavan**[1] · **Ashutosh Sabharwal**[1]

Received: 1 February 2016 / Revised: 24 May 2017 / Accepted: 3 June 2017 / Published online: 7 July 2017

Abstract We study gaze estimation on tablets; our key design goal is uncalibrated gaze estimation using the front-facing camera during natural use of tablets, where the posture and method of holding the tablet are not constrained. We collected a large unconstrained gaze dataset of tablet users, labeled Rice TabletGaze dataset. The dataset consists of 51 subjects, each with 4 different postures and 35 gaze locations. Subjects vary in race, gender and in their need for prescription glasses, all of which might impact gaze estimation accuracy. We made three major observations on the collected data and employed a baseline algorithm for analyzing the impact of several factors on gaze estimation accuracy. The baseline algorithm is based on multilevel HoG feature and Random Forests regressor, which achieves a mean error of 3.17 cm. We perform extensive evaluation on the impact of various practical factors such as person dependency, dataset size, race, wearing glasses and user posture on the gaze estimation accuracy.

Keywords Eye · Gaze estimation/tracking · Dataset · Mobile device · Applications

Electronic supplementary material The online version of this article (doi:10.1007/s00138-017-0852-4) contains supplementary material, which is available to authorized users.

✉ Qiong Huang
qh3@rice.edu

Ashok Veeraraghavan
vashok@rice.edu

Ashutosh Sabharwal
ashu@rice.edu

[1] ECE Department, Rice University, Houston, TX, USA

1 Introduction

Tablets are now a commonplace connected mobile computing device and are in use worldwide for diverse applications. Current user–tablet interactions are mainly enabled by touch and sound. However, gaze is an emerging proxy of the user's attention and intention [11]. Gaze information has the potential to enable a wide array of useful applications on tablets, including (i) hands-free human device interaction, such as using gaze to control the device when certain regions of the screen are hard to reach [30]; (ii) behavior studies, such as using gaze path information for understanding and tracking reading behavior [17]; and (iii) user authentication when gaze-based feature is used as a biometric [23]. In the future, many other applications could be enabled by gaze tracking on tablets.

In this paper, we study gaze estimation on the current generation of tablets, without requiring any additional hardware. Nearly all modern tablets include front-facing cameras. Our approach will be to leverage images from the front-facing cameras for gaze estimation and tracking (gaze estimation at frame rate), thereby making the resulting system suitable for today's tablets.

We adopt an appearance-based gaze estimation approach, since it does not need a specialized hardware setup [4,15,32]. Appearance-based methods find a regression mapping from the appearance of eye region images to the gaze direction, which is then be applied to new unseen eye images. In this way, a regression model could be trained off-line and then loaded on any tablet, estimating gaze using recorded images for any user.

A key challenge in tablet gaze tracking is the ability to robustly handle unconstrained use of tablets. During user–tablet interaction, there is often head motion, hand movement and change of body posture. As a result, shifts in the viewing

angle, changes of distance between the user and the screen, and variations in illumination are possible. Moreover, any useful method should also be capable of tolerating variations in features of subject population, such as eye shape, skin and iris color, and wearing glasses or not. To handle the challenges, the mobile gaze tracking algorithm should be free of three constraints: (i) no constraint on how people use the tablet; (ii) no constraint on what kind of body posture people have when using the tablet; and (iii) no constraint on the user of the tablet.

While unconstrained gaze estimation is practically useful, there exists only a few datasets [16,54] to evaluate the reliability and accuracy of gaze estimation algorithms in an unconstrained mobile setting. As far as we know, none of these works investigated the impact of user-related practical factors, such as body postures, on gaze estimation accuracy.

We study the unconstrained mobile gaze estimation problem in two steps. First, we collected an unconstrained mobile gaze dataset of tablet users from 51 subjects. We name the dataset *Rice TabletGaze dataset*. To the best of our knowledge, this dataset uniquely captures users' gaze appearances under various natural postures and is available online for the research community. While the dataset is collected with one tablet, gaze estimation models trained from this dataset are applicable to other handheld devices, by learned mapping between device specifications such as camera location on the tablet. The dataset consists of video sequences that were recorded by the tablet front-facing camera while subjects were looking at a dot appearing randomly on tablet screen at one of the 35 pre-defined locations. Subjects in the dataset are of diverse ethnic backgrounds, and 26 of them wear prescription glasses. During the data collection process, subject motion was not restricted, and each subject performed four body postures: standing, sitting, slouching and lying. Due to our protocol design, natural and realistic subject appearance variations are captured in the dataset. We obtain a subset of our full dataset, consisting of around 100,000 images from 41 subjects. The subset is labeled with ground truth 2D gaze locations (x and y coordinates on the tablet screen) and used extensively in this paper.

We also extensively evaluate and analyze the impact of factors that could affect gaze estimation accuracy, including person dependency, dataset size, race, prescription glasses and user posture. The studies are based on a baseline algorithm, which estimates a user's gaze given an image recorded by the tablet front camera. The appearance-based baseline algorithm is composed of standard computer vision building blocks. In the algorithm, the eyes in the image are first detected by a cascade eye detector [53], and then, a tight region around the eyes is cropped. A multilevel HoG (mHoG) [24] feature is then extracted from the cropped eye images, and Linear Discriminant Analysis (LDA) is applied subsequently to reduce the feature dimensionality. The final feature is fed into a Random Forests (RF) [3] regressor, which outputs the location on the tablet screen at which the person in the image is gazing. The optimal combination of eye region feature (mHoG) and regression model (RF) is found through performance comparison of five different features and four regressors on the Rice TabletGaze dataset. The baseline algorithm does not estimate explicit head pose information from the images, due to the partial facial visibility of the majority of images, even though it is useful for determining gaze direction. More details are discussed in Sect. 3.2. Then, we conduct extensive experiments on the Rice TabletGaze dataset to study the effect of the aforementioned factors on gaze estimation accuracy. The algorithm is first evaluated on both person-independent and person-dependent training scenarios. Then, we study the gaze estimation performance regarding different training dataset size. Further experiments are specially designed to investigate the impact of race, prescription glasses and user posture. Lastly, we applied the algorithm to videos in the dataset to show continuous tracking results and demonstrated that the error variance can be reduced by using a bilateral filter. An overview of the gaze estimation system setup, the average result and applications of gaze estimation are shown in Fig. 1.

In summary, this paper makes two key contributions:

(i) Rice TabletGaze DataSet: A large gaze dataset was collected in an unconstrained mobile environment, capturing natural and realistic subject appearance variations. This dataset is publicly available at http://www.sh.rice.edu/tablet_gaze.html for research purposes.
(ii) Analysis: A baseline gaze estimation algorithm is presented to investigate the impact of various factors on gaze estimation accuracy. The algorithm achieves a mean error

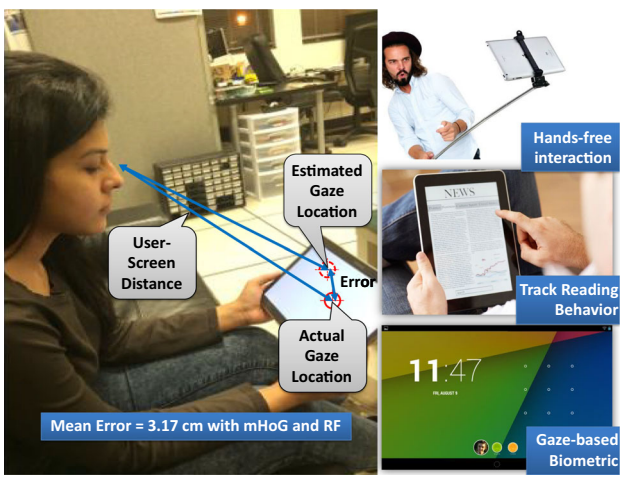

Fig. 1 In this work, we provide a large dataset for unconstrained gaze estimation on tablets. The mean error we obtained is indicated in the figure. A variety of useful applications can be enabled through gaze estimation on mobile device

(ME) of 3.17 cm on the tablet screen. We also demonstrate that person-dependent training scenario resulted in much better accuracy than person independent, showing that person dependency greatly affects gaze estimation performance. The study of the impact of training data size shows that the estimation accuracy can be further increased by collecting more data. We also show that for a large training dataset, dividing the dataset based on racial characteristics and body postures could improve the overall accuracy. However, partitioning the dataset based on whether or not the subject is wearing eyeglasses does not change the algorithm's performance.

2 Related works

We focus on estimating the 2D location on the tablet screen where the user's eyes are focused instead of 3D gaze direction in space. A detailed summary of gaze direction estimation can be found in the following review paper [12].

2.1 Point of gaze estimation for stationary displays

Gaze estimation methods are typically categorized into two main groups: geometry based and appearance based [12]. Geometry-based methods rely on the tracking of certain eye features, such as the iris [15], pupil center [26,34] or Purkinje images [13]. To robustly track the features, those methods require extra infrared illumination source(s), multiple cameras with calibration and sometimes session-dependent personal calibration.

2.1.1 Geometry-based methods

Geometry-based methods utilize explicit 3D eye ball models along with the tracked eye features to estimate the 3D gaze direction. The point of gaze is then found through the intersection of gaze direction and the screen. Based on the pupil center and Purkinje image from one camera and an infrared LED array, a double ellipse fitting mechanism was proposed in [32] to predict the gaze. However, the system required a fixed distance between the display and the user, and head motion was limited to a 4-cm-square area. Meanwhile, an approach free of user calibration was presented in [40]. Two cameras and two point light sources that were calibrated and not colinear were used to find the 3D locations of the cornea and pupil centers. The gaze direction was computed by connecting the cornea center and pupil center. Another approach, proposed in [47], used a single image of one eye to estimate gaze direction. The iris contour in the image was modeled using an ellipse. The ellipse was then back-projected into an iris circle, whose normal was regarded as the gaze direction.

2.1.2 Appearance-based methods

Appearance-based methods [1,20] treat the eye region image or features extracted from the eye region image as a high-dimensional vector and learn a regression mapping model from such vector to the point of gaze (or gaze direction) through labeled training data. Such methods have the potential to be nonintrusive free of calibration and can operate free of external hardware. A variety of regression models were utilized to find the mapping from the eye appearance to point of gaze (or gaze direction) in different works. In [43], eye images were modeled as an appearance manifold. The gaze direction of a new sample was obtained from a linear interpolation of neighboring samples in the manifold model. This method was evaluated only on three subjects with fixed head pose. It used leave-one-image-out cross-validation so test subject's data appeared in the training phase. On the other hand, in [49], a sparse, semi-supervised Gaussian Process Regression model was applied to deal with partially labeled gaze data and realized real-time prediction of gaze direction. The method was evaluated using test images corresponding to unseen gaze locations in the training images. However, there was no description on whether subject's data appear both in the training and testing processes. In addition, there was also no description on whether the data were collected from subjects with a fixed head pose.

In some works [19,21,28,42], 3D head pose information is extracted from images to compensate for head motion, since head pose affects the apparent gaze direction [50]. A two-step scheme was introduced in [19] to estimate gaze direction under free head motion. The method first estimated an initial gaze direction from computed eye features assuming a fixed head pose and then corrected the gaze direction based on head pose rotation and eye appearance distortion. The method was also only evaluated for person-dependent scenario, whereas a person-independent head motion free gaze estimation method was proposed in [28]. A generic gaze model was built upon gaze appearance models of training subjects. Then, a sparse technique was employed to select similar models within the generic model to reconstruct an unseen test subject's frontal eye appearance. The final gaze was determined by transforming the gaze based on head pose estimated from a RGBD camera. In [42], the authors collected a large gaze dataset with multiple head poses. Synthesized eye images were generated through 3D reconstruction of the eye region to provide more data for denser viewing angles. Then, a gaze estimation model was trained using Random Forest on the synthesized images. Finally, both person-independent and person-dependent evaluations were performed on the dataset.

In the meantime, several datasets were released to the public for stationary displays. In [41], Smith et al. introduced a gaze dataset composed of 5880 images from 56 subjects. The

images were recorded from a fixed distance to the subjects in a controlled environment, while they looked at each one of the 21 pre-defined gaze locations. The gaze directions were coarsely arranged in seven horizontal by five vertical angles. Though five horizontal head poses were captured, the vertical head pose was fixed. Sugano et al. [42] collected a large dataset with 64,000 images from 50 subjects. The images have a much denser sampling of gaze angles, with 16 horizontal and 8 vertical gaze directions, and eight head poses. The images were also collected from a fixed distance to the subjects in a controlled environment. A benchmark dataset was proposed in [27] for evaluation of the performance of different gaze tracking/estimation algorithms. The dataset contains videos recorded by both color and depth cameras, and features the variation in head pose, type of gazing target and ambient condition. However, the dataset included data from only 16 subjects, and only three subjects' data were recorded in two different ambient conditions. While all of the above-mentioned datasets captured extensive amounts of head poses and appearances, the experiments were conducted in a tightly controlled manner and do not vary in body posture, which is different from our dataset that is more specifically targeting the mobile usage.

2.2 Point of gaze estimation for mobile displays

Only a few works discussed gaze estimation methods for mobile devices, and most of those works were exploratory, directly applying previously presented methods to mobile devices. In [8], the authors proposed using gaze gestures to control mobile phones, in comparison with gaze dwell duration, and showed the potential to improve gaze tracking accuracy by using gaze gestures. A commercial gaze tracker was utilized to locate the user's gaze location on the phone screen. The change of gaze locations was then converted to gaze gestures. This paper studied only the usability of gaze gestures to control mobile devices based on gaze tracking results, not gaze tracking itself. Nagamatsu et al. [30] adopted the gaze tracking method proposed in [29], utilizing two cameras and two light sources to find the 3D gaze direction on a mobile phone. A one-point personal calibration was used to find the offset between the optical and the visual axes. The system was claimed to work under free hand movement, but there was no quantitative evaluation presented. Kunze et al. [17] implemented an application on mobile tablets and phones to accumulate statistics about user's reading behaviors. They compared the performance of one appearance-based and one geometry-based gaze tracking method and reached the conclusion that both methods are highly dependent on not only the calibration phase but also the position in which the device was held. However, there was also no quantitative evaluation regarding the accuracy of the different methods. In [51], an on-device gaze tracking prototype was implemented using a geometry-based gaze estimation method on an unmodified tablet. The algorithm fitted an ellipse to eye limbus within the region of interest (ROI) detected by eye detectors and found the optical axis through the ellipse normal vector. No user calibration was performed to correct the error between the optical and the visual axes. The optical axis was directly treated as the gaze direction. An accuracy of 6.88° was claimed in the work. However, the method was evaluated only on eight subjects, and subject–tablet distance was fixed in the experiments. Furthermore, the gaze locations included only nine dots on the screen, covering part of the available tablet surface. Recently, Zhang et al. [54] presented a gaze dataset collected under free laptop use with 15 participants. The dataset contains 213,659 images and has 20 gaze locations. An algorithm was also presented in the work utilizing multimodal convolutional neural networks (CNN) to predict gaze direction from head pose and eye appearance. Though laptops are technically a mobile device, they have much less mobility compared to *handheld* devices like tablets and phones. In addition, the statistics from this work showed that the majority of the data were collected during work time, when people would more likely put their laptops on the desk. A major impact from this difference is that the users face is fully visible, while it is certainly not the case for tablets, as is shown in our work. Furthermore, the algorithm presented requires camera calibration and a prebuilt facial shape model. In [16], a large gaze dataset was introduced with 2.5M frames collected from 1450 people. CNN was utilized to estimate a person's gaze directly from an image, without explicitly estimating head pose. This work also considers only the case when the whole face of a user is visible.

Our work studies fully unconstrained handheld mobile device gaze estimation. Our gaze dataset was collected with free subject motion and different body postures, greatly capturing the appearance variations in unconstrained environments. In addition, we made important observations on the Rice TabletGaze dataset that could guide the development of future gaze estimation algorithms. Our study of the impact of practical factors on the algorithm performance such as prescription glasses and body posture, as well as our evaluation of continuous gaze tracking, helps us understand mobile gaze estimation and its practicality.

3 Rice TabletGaze dataset

We created a large publicly available unconstrained mobile gaze dataset, Rice TabletGaze Dataset, to provide data for our study of the unconstrained mobile gaze estimation problem. We designed our data collection experiments to capture unique, unrestrained characteristics in the mobile environment. To this end, we have collected data from 51 subjects,

each with four different body postures. The dataset is also released online to promote future research development of unconstrained gaze estimation methods. While all the data in this paper are recorded with one tablet, one could potentially train a gaze estimation model from this dataset, and the learned model can be used for gaze estimation on other handheld devices through approaches that use transfer learning, domain adaptation or by directly encoding the relative location and resolution of the cameras in the two devices. While we believe this is feasible, it is outside the scope of this paper.

3.1 Data collection

In this research, we used a Samsung Galaxy Tab S 10.5 tablet with a screen size of 22.62×14.14 cm (8.90×5.57 inches). A total of 35 gaze locations (points) are equally distributed on the tablet screen, arranged in five rows and seven columns and spaced 3.42 cm horizontally and 3.41 cm vertically. Example images of the gaze pattern on the tablet screen are shown in Fig. 2. The raw data are videos captured by the front camera of the tablet that was held in landscape mode by the subjects, with an image resolution of 1280×720 pixels.

A total of 51 subjects, 12 female and 39 male, participated in the data collection, with 26 of them wearing prescription glasses; 28 of the subjects are Caucasians, and the remaining 23 are Asians. The ages of the subjects range approximately from 20 to 40 years. An institutional review board (IRB) approval is obtained for the research, and all subjects signed a consent form to allow their data to be used in the research and released online.

During each data collection session, the subject held the tablet in one of the four body postures (standing, sitting, slouching or lying) as shown in Fig. 3 and recorded one video sequence. Each subject needed to conduct four recording sessions for each of the four body postures, so a total of 16 video sequences were collected for each subject. For each recording session, there was no restriction on how the subject held the tablet or how they performed each body posture. The data collection happened in a naturally lit office environment, where only the ceiling lights directly on top of the subjects were turned off to reduce the strong background light in the recorded videos.

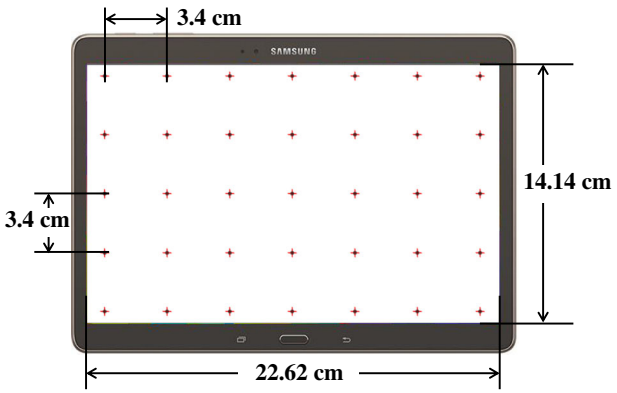

Fig. 2 Gaze locations on the tablet screen. There are 35 (5×7) locations distributed on the tablet screen. In one data collection session, a *dot* appeared at one location at a time and then moved to another location after 3 s. This continued until the *dot* had appeared at all the 35 locations once. The location of the *dot* was randomized among the 35 points

Fig. 3 An example image of the data collection process. In one data collection session, a subject maintains one of the four body postures while gazing at a *dot* on the tablet screen. At the same time, a video of the subject is recorded by the tablet front camera. From *left* to *right*, the subject is standing, sitting, slouching and lying

When a subject started one data collection session, he or she initialized a background recording application on the tablet, so the front-facing camera of the tablet began recording a video of the subject with audio. Then, the subject started to play and watch a video on the tablet. A beep sound notified the beginning of the video, which was also recorded in the video sequence. The recorded sound would be utilized later to locate the time instant in the recorded video when the subject started to watch the video. The video watched by the subjects consists of a dot changing its location every 3 s, and the subject was instructed to focus his/her eyes on the dot the whole time. The subject was free to blink his/her eyes, as it would be uncomfortable to restrain the eye blink in each approximately two-minute long data collection session. To prevent the subject from focusing his eyes to the next gaze point ahead of time (i.e., predicting the dot location), the location of the dot was randomized among the 35 possible points. Sample images from the dataset are shown in Fig. 4.

3.2 Observations on the Rice TabletGaze dataset

In this section, we discuss our observations about facial visibility, body posture and prescription glasses, based on our TabletGaze dataset described in Sect. 3.1.

3.2.1 Observation 1: the entire face may not be visible in most of the image frames

Figure 5 shows an example of full range of facial visibility for the same subject during different data collection sessions.

Fig. 4 Sample images from the Rice TabletGaze dataset. We observe subject appearance variations across different recording sessions. Also, notice that only a fraction of the subject's face is visible in a fraction of the images in the dataset

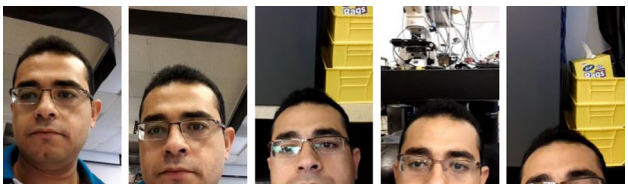

Fig. 5 Example images of five different levels of facial visibility categories. From *left* to *right*, each image represents one of the five following visibility categories: (i) the whole face, (ii) from mouth and above, (iii) from nose and above, (iv) from eyes and above, and (v) where even the eyes are not visible. For clarity of presentation, we have cropped the background

The images vary from full facial visibility to only the subject's forehead being visible. To quantify the extent of facial visibility, we labeled each video in the TabletGaze dataset as belonging to one of the following five categories: (i) the whole face; (ii) from mouth and above; (iii) from nose and above; (iv) from eyes and above; and (v) even the eyes are not visible. For each video sequence, we manually reviewed four images (each image corresponds to one of the four corner gaze locations on the tablet screen) and determined the facial visibility extent of each image. The video sequence is labeled as the majority category of the four images. The statistics based on the above categorization are shown in Table 1.

We observe that the whole face is visible in only 30.8% of all the videos, and the number varies from one posture to another, with sitting being the highest (47%) and lying being the lowest (13.7%). It is clear that in a strong majority of the videos, full facial visibility cannot be assumed.

The extent of facial visibility directly affects the amount of information that can be extracted from the facial region for gaze estimation. For example, head pose information (pitch, yaw and roll angles) along with eye appearance determines a person's gaze [50]. The head pose information can be estimated from the face and can be used in conjunction with eye appearance information to infer gaze estimation. The details were discussed in Sect. 2. However, the bulk of previously proposed head estimation methods [5,37,46] requires the whole face to be visible and is not effective when only part of the face is visible. Due to a lack of robust methods for extracting head pose estimation using partial face visibility, we largely focus on eye region appearance in this paper. We do not use head pose information, even though it is useful in estimating the gaze. However, we did perform preliminary work to incorporate implicit head pose information, such as eye locations in the image frame, as discussed in detail in Sect. 6.

3.2.2 *Observation 2: body posture and facial visibility extent appear to be correlated*

Our starting hypothesis was that there might be a correlation between facial visibility extent and body posture during tablet use. Two main conclusions can be derived from Table 1. First, when seeking a refined amount of information about facial visibility, body posture information can be useful. For example, standing/sitting postures lead to higher probabilities of the face being fully visible, compared to slouching/lying. Intuitively, the observations make sense based on practical experience. Most users tend to rest their tablets on their chest/abdomen when slouching/lying, which reduces chances of seeing the whole face. Although this is beyond the scope of this paper, facial visibility extent could thus potentially be used to roughly estimate the body posture.

Table 1 Statistics on the extent of the visible face region

Posture	Facial visibility				
	Whole (%)	Mouth (%)	Nose (%)	Eyes (%)	No eyes (%)
Standing	39.2	38.2	18.6	4.0	0
Sitting	47.0	27.5	19.1	5.9	0.5
Slouching	23.0	35.8	26.0	13.2	2.0
Lying	13.7	39.7	35.3	7.4	3.9
All body postures	30.8	35.2	24.8	7.6	1.6

Each video in the dataset is labeled as one of the five facial visibility categories. The numbers in the table are percentage of videos. Note that the whole face is only visible in 30.8% of all the videos. Based on these data, we can infer that most of the time the whole face is not visible

Second, if the only objective is to see the eyes, then the eyes are visible in at least 96% of the videos for any posture. Thus, for our proposed appearance-based method discussed in Sect. 4, which relies on the visibility of the eyes only, information about body postures is not essential. However, for methods that may rely on other facial landmarks, the accuracy of gaze estimation could be dependent on the body posture.

3.2.3 Observation 3: prescription glasses can cause reflection, and in many instances, the reflection can be significant

Figure 6 shows examples of eyeglasses reflections from the TabletGaze dataset. Depending on the viewing angle, light source, orientation and coating, there may be no glare from the eyeglasses (left most image in Fig. 6) or very strong glare (right most image in Fig. 6).

To quantify how often reflection happens and how strong the reflection is, we accumulated information on the occurrences and strength of eye glasses reflections in the eye image. We categorized the videos into three broad categories (no reflection, weak reflection and strong reflection) by the same method we used for face visibility categorization. The categorization is done for all the videos of subjects who were wearing glasses, and the statistics are listed in Table 2. We observe that there is visible glasses reflection in half the videos, and in 26.3% of the videos, there is a strong reflection. Reflections with strong intensities could potentially impact the gaze estimation accuracy by i) possibly confusing eye detector used in our algorithms, making it return an erroneous bounding box location around the eye region, and ii) reducing the contrast in some regions of the eye, which in turn makes part of the eye, such as iris or sclera, invisible.

Fig. 6 Example images of different glasses reflection strength. From *left* to *right*, each image represents no reflection, weak reflection and strong reflection, respectively

Table 2 Statistics on eyeglasses reflection strength

Reflection strength	None (%)	Weak (%)	Strong (%)
Number of videos	49.5	24.2	26.3

We can infer that prescription eyeglasses cause reflection in approximately half of the videos

3.3 Sub-dataset Labeling

The total amount of raw data collected is $51 \times 16 = 816$ video sequences. However, a portion of the data is not usable for three reasons: (i) the transition from one gaze point to the next and loss of concentration of subjects produce image frames with inconsistent gaze labels; (ii) the eye detector failure in some conditions causes missing data; and (iii) involuntary eye blinks and large motion blur result in images without useful gaze information from the eyes. Because of these three reasons, we prune the raw data and obtain a sub-dataset of 41 subjects to be used in our experiments. Below, we explore the three reasons in more detail and describe how we filter out the unusable data.

We first remove images with inconsistent gaze labels. We extract only the video chunk that corresponds to 1.5–2.5 s after the time the dot appears at a new location to remove the time for subjects to refocus. Since it is unavoidable that sometimes the subject loses concentration during a data collection session, the gaze label of parts of the corresponding video data can be mismatched. For the 35 video chunks extracted from each video sequence, we visually inspect whether there is a gaze drift for more than five video chunks and, if so, abandon the data from the whole video sequence. Since it is hard to determine the true gaze location just by looking at one stand-alone image, we extract one eye region image for each gaze point and enhance the contrast of the image to compensate for the low illumination scenario. By compar-

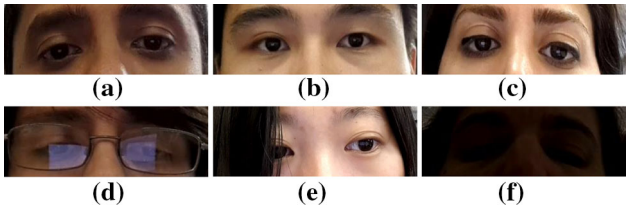

Fig. 7 Eye detection fails in some scenarios. *Top* row of images, **a**–**c**, shows correct eye detection cases. *Bottom* row of images represents eye detection failure cases, including **d** strong glasses reflection, **e** hair occlusion over an eye and **f** poor illumination

ing the relative location of the iris and openness of the eyes among 35 gaze locations, we are able to identify each gaze drift occurrence and calculate the total number of gaze drifts.

We then remove images with eye detector failures. For each video chunk of time duration 1 s, the number of frames contained is between 15 and 30 due to the variable video recording rate of the front camera. An important step for automatic estimation of gaze through images is to detect the eye region using an eye detector, which fails in conditions; for example, eyes are not visible in the image frame, strong reflection from prescription glasses, occlusion from hair, poor illumination, and so on. Images with eye detector failures are removed, resulting in small data size for certain subjects. Examples of eye detection success and failure cases are shown in Fig. 7.

Another source of images without useful information is the involuntary blinking and occasional large motion of the subjects during the data collection stage. Since the images of closed eyes and blurred eye regions are undesired, for image frames within each video chunk corresponding to one gaze direction, we extract five images with lower mean intensity value and higher mean Laplacian of Gaussian (LoG) value. We do this because images of closed eyes will have higher mean intensity value given the disappearance of the dark pupil, and a blurred eye region image will have a lower mean LoG value because motion blur weakens the edge information in the image. Even though some video chunks do not contain closed eye images, we still extract five image frames to guarantee a similar number of data samples for each gaze point.

This extensive data selection process removes most of the unusable images. The tiny fraction of bad images that escape this procedure is treated as noise.

4 Baseline gaze estimation algorithm

In this section, we describe the baseline algorithm that leverages well-known machine learning processing modules. The baseline algorithm is similar to the gaze estimation method used in [38], but it is optimized for our dataset. The building blocks of the algorithm are shown in Fig. 8. The estimation of gaze from an image consists of three parts: preprocessing, feature extraction and regression estimation. The preprocessing part involves image normalization (e.g., scaling) so the eyes from different images can be directly compared. For feature extraction and regression, we select the feature and regressor combination that yields the best result on our TabletGaze dataset. We tested five features including contrast normalized intensities, Laplacian of Gaussian (LoG), Local Binary Patterns (LBP) [33], Histogram of Oriented Gradients (HoG) [6] and multilevel HoG (mHoG) [24]. We utilized four regressors, namely k-Nearest Neighbors (k-NN), Random Forests (RF) [3], Gaussian Process Regression (GPR) [48] and Support Vector Regression (SVR) [9].

4.1 Preprocessing

The first step is to preprocess the input images, which have a resolution of 1280 × 720 pixels. An example of the preprocessing step is displayed in Fig. 9. We first apply two Harr feature CART tree-based cascade detectors [53], one trained for left eye and one for right eye, to locate image patches that include potential left and right eye regions. A sample output of the detectors is shown in Fig. 9. False positive bounding boxes from the detectors are rejected by (1) empirically establishing a threshold for the size of the box to remove small false positive patches, such as the nostril detected in Fig. 9, and (2) enforcing coarsely symmetric locations of the bounding boxes returned by the left and right eye detectors (to compensate for head tilt where eyes are not totally symmetric) to remove stand-alone false positive patches, such as the mouth detected in Fig. 9. The eye region bounding box sizes vary for different images, so their sizes are scaled to 100 × 100 pixels. The detected bounding box contains a large area including the eye brows, which is not informative about gaze, so we crop a tight box around the eye to procure the final eye image. The pupil center is coarsely located at one half horizontally and two thirds vertically of the bounding box, given that the aforementioned eye detector was trained with eye images of this geometry. We crop 15 pixels from the top and bottom around the pupil center to form the final eye image, which covers the eye region tightly for most subjects. The horizontal dimension is untouched since the eye width varies widely among different subjects. As a result of the aforementioned operations, the final eye image size becomes a fixed 30 × 100 pixels for each eye across all images. A few sample images of the cropped eye regions are shown in Fig. 10. Blinks cause the gaze estimation algorithm to produce incorrect predictions and need to be removed. To detect blinks, the algorithm looks for changes in the mean pixel intensity of the eye region over time. The algorithm takes advantage of the fact that when an eye blink occurs, the continuous disappearance and reappearance of the

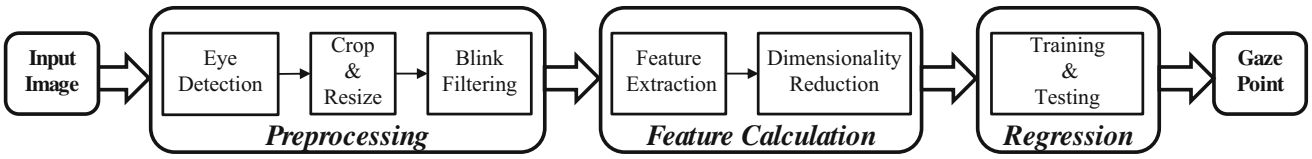

Fig. 8 Automatic gaze estimation algorithm. The input to the algorithm is an image of the user recorded by the tablet front camera. The output is the location on the tablet screen at which the user is looking. The estimation of gaze from an image consists preprocessing, feature extraction and regression estimation

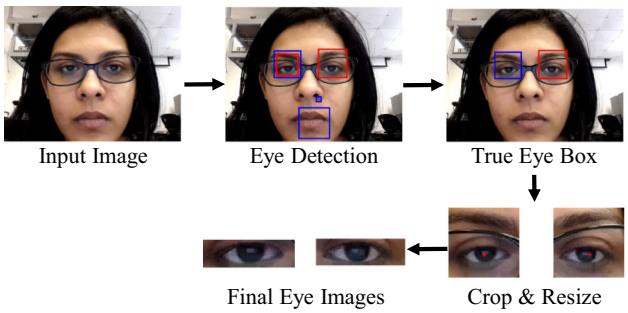

Fig. 9 Example images in the preprocessing phase. Firstly, two eye detectors are applied to detect potential left and right eye regions. The *blue* bounding boxes denote the output of left eye detector, while the red bounding boxes denote the output of right eye detector. In the example image, we can observe false positive image patches around the nostril and mouth, which are removed to find the true eye region. Then, only a tight region around the eyes is used to avoid the ambiguity caused by eye brows and facial expressions (color figure online)

Fig. 10 A few sample images of the final extracted eyes data. Each row of eye images comes from one subject. We observe that after the preprocessing step, the eyes are tightly cropped

dark pupil result in an increase and then decrease in the mean pixel intensity. The mean is taken over 20 consecutive frames, which is usually less than the time length between two consecutive blinks. By inspecting the video sequences, we found that an eye blink usually lasts around 4–6 frames. Therefore, when a blink is detected, we skip six frames around the peak frame.

4.2 Feature calculation

Following eye extraction, we next find features. Feature calculation includes two steps: feature extraction and dimensionality reduction.

Feature Extraction: The accuracy of gaze estimation greatly depends on the feature we choose. To ensure the baseline algorithm achieved a state-of-the-art result, we chose to evaluate the performance of five popular features: (1) contrast normalized pixel intensities; (2) LoG; (3) LBP; (4) HoG; and (5) mHoG feature. The first proposed feature, contrast normalized pixel intensities, is the simplest feature of the five; it converts pixel values into the feature vector after normalization to account for variations in illumination. LoG convolves each eye image with a LoG filter and concatenates the returned vector to enhance eye contour and remove person-dependent eye region texture information. LBP and HoG have been proved by many works as powerful features [7]. LBP captures image texture information, while HoG retrieves local shape and orientation information. As a variant of HoG, multilevel HoG (mHoG) is formed by concatenating HoG features at different scales. The block scales utilized in this paper are the same as presented in [25].

Dimensionality Reduction: Features obtained in the feature extraction phase suffer from being high dimensional and compromised by noise. We overcome these problems by mapping the features to a lower-dimensional space. In this work, we applied Linear Discriminant Analysis (LDA) to reduce the feature dimensionality. LDA maps the data to a lower-dimensional space where the inter-class scatter to intra-class scatter ratio is maximized. Finding the projection vector requires computation of the inverse of intra-class scatter. The intra-class scatter matrix suffers from a singularity problem when the number of data samples per class is smaller than the number of features. Regarding this, we applied Principal Component Analysis (PCA) to the original feature data to reduce its dimension. The dimension is reduced to no smaller than the number of observations per gaze point. Then, we apply LDA to the already reduced data to obtain a final feature vector. Note that LDA is applied on both training and test data, but the LDA projection vector is obtained from the training data only, since LDA requires known data label and test data label should be unknown, whereas PCA projections are learned from both train and test data, since the learning does not require known data labels. Given input data of feature length C, the output data of LDA will have a length of $C - 1$. In our dataset, we have gaze data corresponding to 35 gaze locations/classes, so the final data after the LDA operation have a feature length of 34.

4.3 Regression

Finally, after computing the final feature vectors, the data are fed into a regression model. The gaze labels of the data include two parts: the horizontal and vertical (x and y) coordinates on the tablet screen. We trained a separate regressor for the horizontal and vertical gaze locations, respectively. Then, the output from the two regressors is combined as the predicted 2D gaze location on the tablet screen. In our work, we experimented with the four different models mentioned earlier. k-Nearest Neighbors (k-NN) assigns the average of the output of the k-Nearest Neighbors in the training data to a new observation; we chose $k = 3$ in our experiments.

Random Forests (RF) are a set of weak binary tree regressors. Each tree in the forest is grown by randomly bootstrapping samples and each binary split of the tree is grown by randomly selecting a subset of the features. For regression RF, the output of a new input is given by the average of the output of each tree in the forest. RF has previously been used in gaze estimation papers and shows strong performance [42,52]. In our experiments, we used 100 trees.

Gaussian Process Regression (GPR) models the regression problem as a Gaussian process and estimates the output of a new observation by taking the conditional probability over the training data. The advantage of GPR is that it returns not only the estimate of the output, but also the confidence interval of the estimate. However, traditional GPR has a complexity of $O(N^3)$ for an input data samples size N [36], which makes it computationally infeasible for a large dataset, such as the over 100,000 samples in our data. In our experiments, we used fully independent training conditional (FITC) approximation [31], a sparse GPR method which claims to achieve similar accuracy as GPR, to reduce the running time. Even with the faster FITC approximation, we could only manage to evaluate on 15 subjects with a reasonable computing time using threefold cross-validation.

Support Vector Regression (SVR) utilized the well-known "kernel trick" to project data into a higher-dimensional space where a linear regression function can effectively fit the data. A nonlinear kernel can transform a nonlinear regression problem in the original data space into a linear one in the new space. In our experiment, we employed the popular nonlinear radial basis function (RBF) kernel. The performance of SVR depends highly on the model parameters, which are usually obtained through a coarse to fine grid search process. Given a data sample size N, SVR has a training time complexity of $O(N^3)$ [45], which greatly limits its scalability to large datasets. In our experiments, we evaluated SVR on the subset of 15 subjects that was used in GPR evaluation. The evaluation was also conducted using threefold cross-validation.

5 Results and analysis

5.1 Error metric

Previous works on gaze estimation employed an angular error to evaluate the quality of gaze estimation. The angular error is computed by taking the arctangent of the ratio between the distance from the subject's eyes to the screen center and the distance of the gaze point from the screen center. However, in the mobile environment, the distance between the screen and the user is highly variable, so it is not possible to reliably calculate the angular error. For our work, since we have the ground truth gaze labels (2D location on the table screen) for all data, we define the estimation error of one data sample as the Euclidean distance between the predicted 2D gaze point location and the actual 2D gaze point location on the tablet screen. The final error is reported as the mean error (ME) over all data samples.

5.2 Comparisons for different features + regressors

In Table 3, we first summarize the performance of each feature and regressor as described in Sects. 4.2 and 4.3. The entries in Table 3 are the MEs (in cm) across around 100,000 images from 41 subjects using the leave-one-session-out cross-validation. The details are described in Sect. 5.3. The columns of Table 3 represent different features. The features are listed in order of increasing complexity, and this trend can be seen in the table—estimation accuracy generally increases as feature complexity increases regardless of the classifier used. Note that the complex texture feature, LBP, performs no better than the simple edge feature LoG and delivers far inferior performances as complex shape and edge features, HoG, and mHoG. We hypothesize that the lack of performance improvement is because the shapes and edges, such as those from the limbus and sclera, communicate more information about the gaze location than texture does. Note that, mHoG and HoG achieve the best results and mHoG performs slightly better than HoG, while other features yield far worse results. Moreover, the computation of mHoG feature is fast due to the utilization of integral histograms [35].

The rows of Table 3 represent different regressors. We notice that the best two results both come from the RF regressor. In addition, RF provides fast prediction results and thus has been widely adopted in real-time systems [10]. In our experiments, we actually found the results were reasonably stable when using more than 20 trees; we used 100 trees to further improve accuracy.

Overall, mHoG and RF achieve the lowest error of 3.17 ± 2.10 cm, as listed in Table 3. A few example images with high estimation error are shown in Fig. 11. Even considering the computational complexity (e.g., for real-time applications), mHoG and RF are still recommended for their

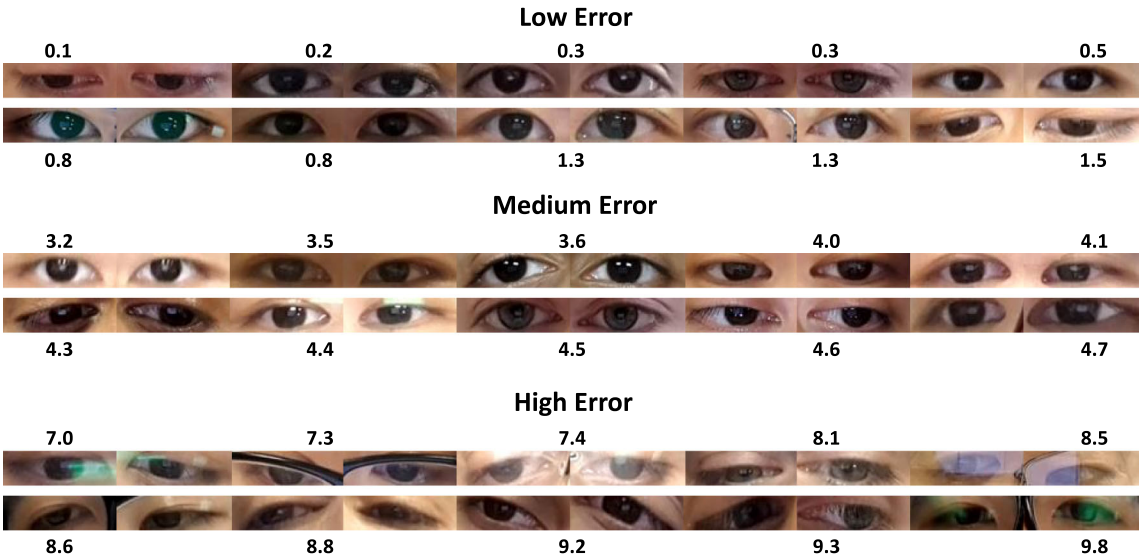

Fig. 11 Example eye images with different gaze estimation errors. In the figure, we show ten pairs of eyes for low, medium and high estimation errors (in cm) using mHoG + RF gaze estimation algorithm. We can observe that factors such as erroneous eye region cropping, long eyelashes, strong reflections from prescription glasses, glass frames, rotated eyes and motion blur can reduce estimation accuracy

Table 3 Mean error (cm) for each feature and regressor combination

Regressors	Features				
	Raw pixels	LoG	LBP	HoG	mHoG
k-NN	9.26	6.45	6.29	3.73	3.69
RF	7.20	4.76	4.99	3.29	**3.17**
GPR[a]	7.38	6.04	5.83	4.07	4.11
SVR[b]	×	×	×	×	4.07

Note that the combination of mHoG feature and RF regressor achieved the lowest error, which is marked in bold

[a] Due to training time complexity constraint, GPR is evaluated using threefold cross-validation on data of 15 subjects, which is essentially leave-five-subjects-out

[b] SVR is evaluated only on the optimal feature, which is mHoG. The evaluation process is conducted in the same way as GPR

relatively fast computation. This is our chosen baseline algorithm for the experiments in the following sections.

5.3 Person-dependent and person-independent performance comparison

Here, we study the influence of person dependency, not session dependency, on algorithm performance. The analysis of session dependency is not useful because in daily use, a person's appearance can vary widely between sessions. In a person-dependent model, the performance is evaluated using leave-one-session-out cross-validation on the data from the same person (each person has 16 video recording sessions). For one subject, each time the images from one session (one recorded video) are left out as the test data, and the images from the rest sessions of this subject are used to train a regressor. Then, we repeat the evaluation process for all the rest subjects. The final estimation error is equal to the average of all test data's estimation errors. In the person-independent model, a leave-one-subject-out cross-validation is employed. In each one of the 41 evaluation rounds (the TabletGaze dataset includes 41 subjects), the regressor is trained on data from 40 subjects and tested on the remaining one subject, and then, the final results are obtained by averaging the estimation errors of all the images from the 41 subjects.

Figure 12a shows the estimation error histograms over all the images in the sub-dataset. We observe that for person-dependent training scenario, the estimation errors aggregate near lower values compared to person-independent training scenario. The observation implies that for the person-dependent training scenario, the estimation error is lower than that in the person-independent training scenario. The numerical MEs over all samples in the sub-dataset are shown in Fig. 12b for the two training scenarios. This result is expected because the regressor will have better generalization power for images from the same person, due to the stronger similarity between the images.

We also present the stand-alone horizontal and vertical errors (x and y coordinates on the tablet screen), in addition to the overall/combined ME for both person-dependent and person-independent training scenarios in Fig. 12b. The horizontal and vertical errors are both evaluated using mean absolute error (MAE) to avoid the cancelation of positive and negative errors. Unidirectional gaze estimation might be useful for applications that requires only information from a singular direction, such as Web-page scrolling. We observe that the horizontal and vertical errors are similar, showing

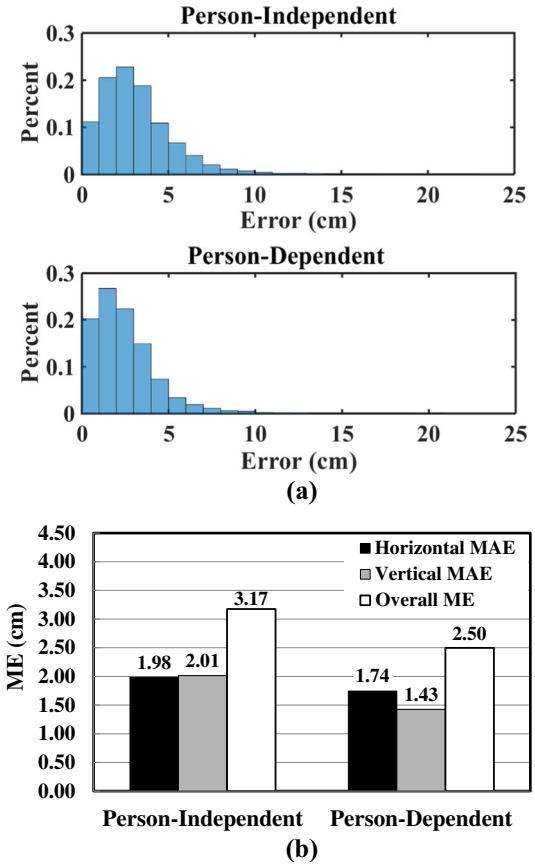

Fig. 12 Person-independent and person-dependent training performance comparison. In person-independent training, leave-one-subject-out cross-validation was utilized to evaluate the algorithm, while in person-dependent training, leave-one-session-out cross-validation was employed. **a** Error histograms. **b** Bar plot comparison

that the horizontal and vertical regressors have similar predictive powers.

5.4 Effect of training data size

In this section, we study the impact of training data size on the estimation accuracy of the baseline algorithm. We randomly select groups of different number of participants for evaluation. We experiment with groups of different sizes K, where K is within the range [2, 41]. For each group, we perform leave-one-subject-out cross-validation, so in each training round we use $K - 1$ subjects' data. Since we are randomly selecting a subset of data from the whole data, we repeat the same process five times and average the final reported errors to reduce bias.

The results are presented in a semi-log plot as shown in Fig. 13. As the size of the training group increases, the estimation error decreases monotonically. The monotonically decreasing relationship suggests that if we use more training subjects, we can further improve estimation accuracy.

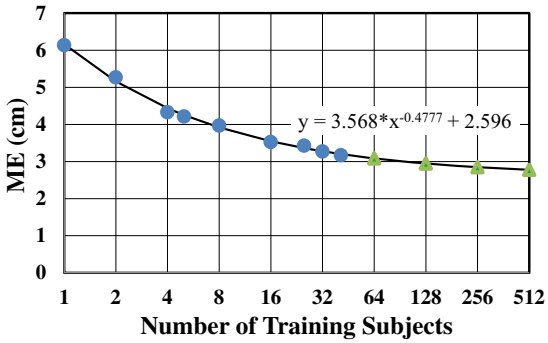

Fig. 13 Effect of training data size on the gaze estimation accuracy of the baseline algorithm. The *round circles* are results obtained in the experiment, and a *line* is fitted to the data points. The *triangles* are data points derived through extrapolation on the *fitted line*. We can see that the ME decreases monotonically as the number of training subjects grows larger, indicating that more data could improve the performance further

5.5 Eyeglasses, race and posture

We validate whether dividing the dataset into groups based on person-related factors and training a separate regressor for each group would further reduce the estimation error. Our hypothesis is that the eye appearance variations caused by factors other than gaze can be reduced within each group. Previous works on head pose estimation [14,55] and face detection [18] demonstrated improved accuracies by dividing the data into groups and training a regressor/detector for each group. At the same time, we also examine the impact of each factor on gaze estimation accuracy. Due to a lack of sufficient data in some of the categories, for example, we have only six subjects who are Caucasians and wearing glasses, we could not perform controlled tests to study the impact of each independent factor. Nevertheless, we can still gain some initial understanding of the impact of the three factors on the performance of the gaze estimation algorithm. Three factors are considered in our study: eyeglasses (wearing eyeglasses or not), race (Caucasian or Asian) and body posture (standing, sitting, slouching or lying). Three experiments are conducted for each factor.

5.5.1 Eyeglasses

We first discuss the impact of eyeglasses. A diagram of the experiment design for the three experiments is shown in Fig. 14. The dataset is first divided into two groups: Group 1 is wearing glasses and Group 2 is not. In the first experiment, leave-one-subject-out cross-validation is evaluated on the data of each group separately, and the estimation errors are obtained for each group. In our data, there is an unequal number of subjects within each group. To solve this problem, suppose Group 1 has M subjects and Group 2 has N subjects, where M is larger than N. Then, we randomly select N sub-

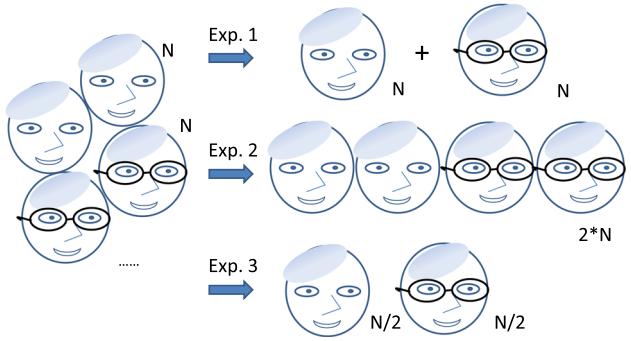

Fig. 14 Diagram of the design for the three experiments studying the factor of prescription glasses. In Exp. 1, the dataset was partitioned into two groups of wearing glasses (Group 1, N subjects) and not wearing glasses (Group 2, N subjects), and training and testing were done separately for each group. In Exp. 2, the leave-one-subject-out cross-validation was conducted on all data ($2 \times N$ subjects), but the ME was separated for each group. In Exp. 3, we combined data of half of the subjects from Group 1 and half from Group 2 and conducted training and testing within the combined data

jects from Group 1 and run Exp. 1. We repeat the experiment five times and average the ME for Group 1 to reduce bias caused by random selection. The second experiment is conducted on data from both groups using leave-one-subject-out cross-validation. The estimation error is separated depending on whether the test subjects are wearing glasses or not. In Exp. 1, the number of training subjects is smaller than the number of training subjects in Exp. 2 due to data partitioning. We can infer that this size discrepancy will have a negative impact on the estimation accuracy, as discussed in Sect. 5.4. To mitigate the effects of training data size, in Exp. 3 we choose the same training data size as in Exp. 1. We randomly select $N/2$ subjects from Group 1 and $N/2$ subjects from Group 2 and combine the data in Exp. 3. The evaluation process is done using the same method as in Exp. 2. Exp. 3 is also repeated five times to reduce the bias caused by the random selection of training subjects.

The results are shown in the first bar plot of Fig. 15. As we can observe from the bar plot, the ME of the group of wearing glasses is larger than the group of not wearing glasses for all the three experiments. We can also observe that in Exp. 1, the ME increases around 0.4 cm for the group of wearing glasses compared to the group of not wearing glasses. These observations means that wearing glasses has a negative impact on gaze estimation accuracy. We can tell from the bar plot that for each group, the ME of Exp. 1 is higher than the ME of Exp. 2. The increase in error means that partitioning the data does not improve accuracy when we have limited number of training subjects. We obtain similar ME for Exp. 1 and 3, showing that partitioning the data based on the factor of glasses does not have a significant impact on estimation accuracy when we have sufficient training data. The factor of glasses does not affect estimation accuracy most likely because sometimes the reflection from glasses is not strong and does not introduce much noise in the eye images.

5.5.2 Race

We utilize the same approach as in Sect. 5.5.1 to design the three experiments to study the impact of racial characteristics. The second bar plot of Fig. 15 shows the results. We obtain quite different MEs for the group of Caucasians and the group of Asians, which tells that the factor of race impacts the performance of the gaze estimation algorithm. We also notice that for each individual group, the ME of Exp. 1 is higher than the ME of Exp. 2 while the ME of Exp. 1 is lower than that of Exp. 3. We can infer that that partitioning the data does not improve accuracy when we have limited number of

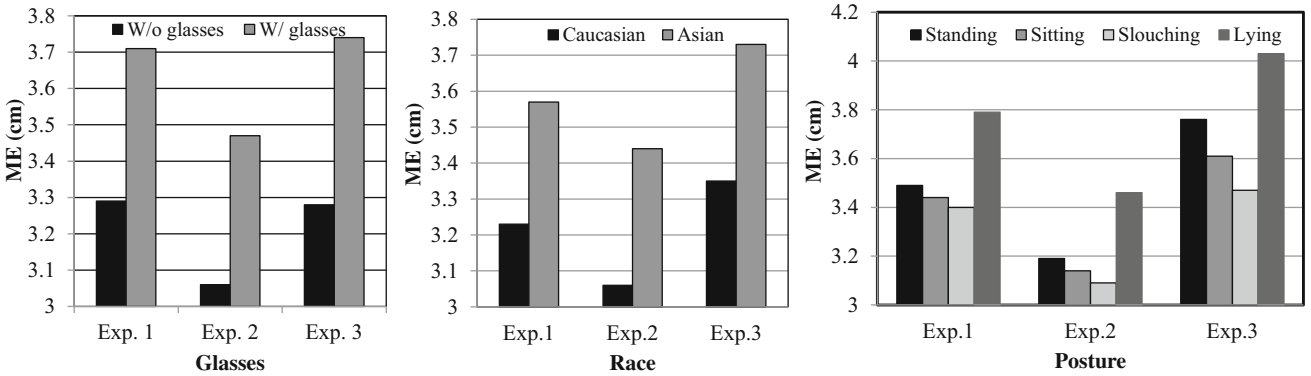

Fig. 15 Study of whether partitioning the data based on person-related factors would reduce estimation error. The error obtained in Exp. 2 is lower than that in Exp. 1 for all the three factors. It means that when we have limited training subjects, data partition increases the estimation accuracy. The error obtained in Exp. 3 is higher than that in Exp. 1 for racial characteristics and body posture and almost the same for wearing glasses or not. It infers that when we have sufficient training subjects, data partition based on the factor of race and boy posture improves the estimation accuracy, while the factor of glasses does not significantly impact the result

training subjects. Moreover, when we have a large amount of training data, dividing the data based on race improves accuracy because people within the same racial group have similar eye shapes.

5.5.3 Body posture

For studying the impact of body posture (standing, sitting, slouching or lying), the data partition is performed differently. As described in Sect. 3.1, we have four subsets of data for each posture for each subject. We partition the dataset into 4 groups, each group containing data of one body postures from the same subjects. For each subject, the data size for each body posture may be unequal due to occasional unusable data. To reduce the effect of unequal dataset size, we choose a subset of 29 subjects from the TabletGaze dataset where the amount of each subject's data for each body posture is almost equal. Then, we perform the three experiments in the same way as described earlier in this section.

The results are shown in the third bar plot of Fig. 15. We notice that the MEs of the standing, sitting and slouching groups are quite similar, while the ME for the group of lying is the highest. One reason for the high error of the group of lying is that people have more varied head pose and way of holding the tablet when they are lying. We also notice that for each individual group, the ME of Exp. 1 is higher than the ME of Exp. 2 while the ME of Exp. 1 is lower than that of Exp. 3. We can infer that that partitioning the data does not improve accuracy when we have limited number of training subjects. Moreover, when we have a large amount of training data, dividing the data based on body posture improves accuracy because people might have similar head poses when they are doing the same posture.

5.6 Continuous gaze tracking from videos

We apply the baseline algorithm to continuously track user's gaze on videos in the sub-dataset. Initially, we directly estimate user's gaze in the videos on a frame-by-frame basis. When implementing a continuous gaze tracking system, temporal information can be utilized to further reduce gaze prediction errors. A temporal bilateral filter can be applied on consecutive gaze estimations to reduce the miniature fluctuation of neighboring gaze estimations caused by model noise and preserve the large gaze shifts due to change of fixation location. Some example images of the continuous gaze tracking based on the baseline algorithm and the effect of bilateral filter are displayed in Fig. 16. We can observe that for each gaze location in the image, the gaze estimations are close to the ground truth gaze locations, and the errors are less than 3.4 cm (the distance between two cross-stiches), which conforms to the ME of 3.17 cm. We also notice that after applying the bilateral filter, fluctuations of the gaze estimations for each

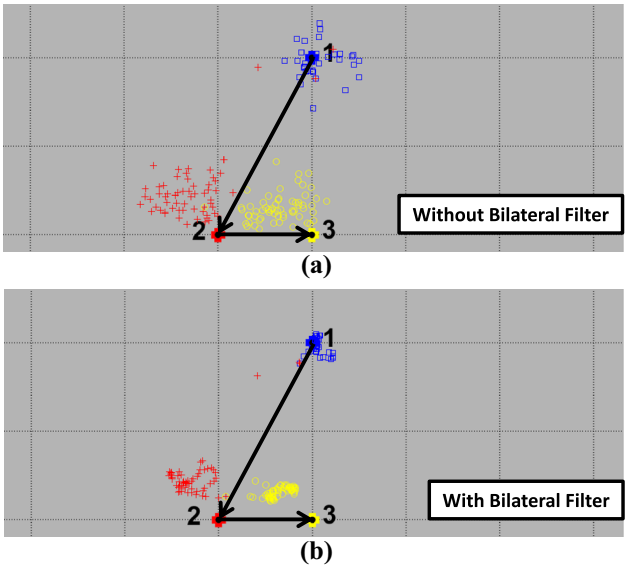

Fig. 16 Continuous gaze tracking demonstration. Each image shows a part of the tablet screen, and the cross-stitches of the *grid lines* represent the 35 potential ground truth gaze locations. In **a**, three ground truth gaze locations and the color-coded predicted locations are shown for a single subject from our dataset. Each location is showed in the sequence indicated by the *arrows*. The distance between the predicted gaze locations and the true gaze location is within the distance between two cross-stiches (3.4 cm). In **b**, the predicted locations are passed through a bilateral filter; the fluctuations of the predictions are reduced by the filter

ground truth gaze location are decreased. Meanwhile, temporal eye center location information can be collected, and the change of subsequent eye center locations can be used to correct gaze estimations. For example, sometimes a user naturally moves his/her head from left to right when he/she is looking from left to right on the screen. Along with the head movement, the eye center location would also shift to the right. This shift of eye location can thus be utilized to correct neighboring gaze estimates so the predicted gaze location also changes accordingly.

6 Discussion and conclusion

All of the evaluations of the algorithm are conducted using MATLAB running on a desktop computer. The code is not optimized to run in real time on a tablet. However, the algorithm could possibly be improved to execute in real time. When implementing the algorithm on a tablet, the RF regressor can be pre-trained off-line and loaded onto the device. The eye region could be tracked using common visual tracking methods, such as Kanade–Lucas–Tomasi tracker [22,39,44] with proven real-time performance [2]. The computation of the mHoG feature from an eye image is fast, due to the introduction of integral histograms [35]. Regression prediction

using the RF model has also shown to be running in real time [10].

As we discussed earlier, direct 3D head pose information cannot always be obtained for the mobile environment due to partial facial visibility in some cases. Thus, an explicit 3D head pose is not utilized in this work. As the increasing popularity of dual-camera mobile devices, we could leverage the information from two front-facing cameras to obtain head pose for gaze estimation in the future. Here we discuss an exploratory experiment regarding incorporating implicit head pose information. Head pose information is correlated with features such as the location of the eye center in the image frame and the size of the eyes, which can be extracted as alternatives to exact head pose angles. To utilize this information, we design a feature vector composed of the following features: the x and y coordinates of the left and right eyes, eye sizes (width and height of the eye bounding boxes) and the x and y location difference between the left and right eyes. This feature vector has a length of 10 and is combined with the LDA-reduced mHoG feature as an input to the RF regressor. The data are also evaluated using leave-one-subject-out cross-validation, and we obtain a ME of 3.10 ± 2.07 cm. There is no significant improvement compared to the 3.17 ± 2.10 cm ME when we do not use the eye location information. This means directly adding these features does not result in significantly improved estimating accuracy. A future direction of this work could be focused on designing a new scheme to appropriately and productively incorporate head pose information.

In conclusion, this work presented and studied the unconstrained mobile gaze estimation problem in two major steps. Firstly, a large dataset was collected in an unconstrained environment. The dataset is designed to explore the variation in subject appearances in an unconstrained environment by including four different postures and recording the data in videos. Three observations were made on the dataset, including facial visibility, posture and glasses reflection, which provide a deeper understanding of the challenges present in the mobile environment. We implemented a baseline gaze estimation algorithm, which achieves a ME of 3.17 ± 2.10 cm on the tablet screen. This result is useful enough for applications that do not require high gaze estimation accuracy, such as detecting whether an user is looking at the advertisement at the bottom of an app. Utilizing the baseline algorithm, we investigated the impact of several practical factors, including person dependency and eye glasses, on gaze estimation accuracy. These studies give insight to fellow researchers on investigating new ways to further improve gaze estimation.

Acknowledgements We acknowledge the support from National Science Foundation (NSF) Grants NSF-IIS: 1116718, NSF-CCF:1117939 and NSF-CNS:1429047. We would further like to thank all the participants in the dataset for volunteering and allowing their data to be released.

References

1. Baluja, S., Pomerleau, D.: Non-intrusive gaze tracking using artificial neural networks. Tech. rep, DTIC Document (1994)
2. Benfold, B., Reid, I.: Stable multi-target tracking in real-time surveillance video. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pp. 3457–3464. IEEE (2011)
3. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
4. Brolly, X.L., Mulligan, J.B.: Implicit calibration of a remote gaze tracker. In: Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on, pp. 134–134. IEEE (2004)
5. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models-their training and application. Comput. Vis. Image Underst. **61**(1), 38–59 (1995)
6. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 1, pp. 886–893. IEEE (2005)
7. Dollar, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: an evaluation of the state of the art. IEEE Trans. Pattern Anal. Mach. Intell. **34**(4), 743–761 (2012)
8. Drewes, H., De Luca, A., Schmidt, A.: Eye-gaze interaction for mobile phones. In: Proceedings of the 4th International Conference on Mobile Technology, Applications, and Systems and the 1st International Symposium on Computer Human Interaction in Mobile Technology, pp. 364–371. ACM (2007)
9. Drucker, H., Burges, C.J., Kaufman, L., Smola, A., Vapnik, V., et al.: Support vector regression machines. Adv. Neural Inf. Process. Syst. **9**, 155–161 (1997)
10. Fanelli, G., Gall, J., Van Gool, L.: Real time head pose estimation with random regression forests. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pp. 617–624. IEEE (2011)
11. Frischen, A., Bayliss, A.P., Tipper, S.P.: Gaze cueing of attention: visual attention, social cognition, and individual differences. Psychol. Bull. **133**(4), 694 (2007)
12. Hansen, D.W., Ji, Q.: In the eye of the beholder: a survey of models for eyes and gaze. IEEE Trans. Pattern Anal. Mach. Intell. **32**(3), 478–500 (2010)
13. Hennessey, C., Noureddin, B., Lawrence, P.: A single camera eye-gaze tracking system with free head motion. In: Proceedings of the 2006 Symposium on Eye Tracking Research & Applications, pp. 87–94. ACM (2006)
14. Huang, J., Shao, X., Wechsler, H.: Face pose discrimination using support vector machines (svm). In: Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on, vol. 1, pp. 154–156. IEEE (1998)
15. Kim, K.N., Ramakrishna, R.: Vision-based eye-gaze tracking for human computer interface. In: Systems, Man, and Cybernetics, 1999. IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on, vol. 2, pp. 324–329. IEEE (1999)
16. Krafka, K., Khosla, A., Kellnhofer, P., Kannan, H., Bhandarkar, S., Matusik, W., Torralba, A.: Eye tracking for everyone. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2176–2184 (2016)
17. Kunze, K., Ishimaru, S., Utsumi, Y., Kise, K.: My reading life: towards utilizing eyetracking on unmodified tablets and phones. In: Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication, pp. 283–286. ACM (2013)
18. Li, S.Z., Zhu, L., Zhang, Z., Blake, A., Zhang, H., Shum, H.: Statistical learning of multi-view face detection. In: Computer Vision ECCV 2002, pp. 67–81. Springer (2002)

19. Lu, F., Okabe, T., Sugano, Y., Sato, Y.: Learning gaze biases with head motion for head pose-free gaze estimation. Image Vis. Comput. **32**(3), 169–179 (2014)
20. Lu, F., Sugano, Y., Okabe, T., Sato, Y.: Adaptive linear regression for appearance-based gaze estimation. IEEE Trans. Pattern Anal. Mach. Intell. **36**, 2033–2046 (2014)
21. Lu, F., Sugano, Y., Okabe, T., Sato, Y.: Gaze estimation from eye appearance: a head pose-free method via eye image synthesis. IEEE Trans. Image Process. **24**(11), 3680–3693 (2015)
22. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: International Joint Conference on Artificial Intelligence, vol. 2, pp. 674–679. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA (1981)
23. Maeder, A., Fookes, C., Sridharan, S.: Gaze based user authentication for personal computer applications. In: Intelligent Multimedia, Video and Speech Processing, 2004. Proceedings of 2004 International Symposium on, pp. 727–730. IEEE (2004)
24. Maji, S., Berg, A.C., Malik, J.: Classification using intersection kernel support vector machines is efficient. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pp. 1–8. IEEE (2008)
25. Martinez, F., Carbone, A., Pissaloux, E.: Gaze estimation using local features and non-linear regression. In: Image Processing (ICIP), 2012 19th IEEE International Conference on, pp. 1961–1964. IEEE (2012)
26. Merchant, J., Morrissette, R., Porterfield, J.L.: Remote measurement of eye direction allowing subject motion over one cubic foot of space. IEEE Trans. Biomed. Eng. **4**, 309–317 (1974)
27. Mora, K.A.F., Monay, F., Odobez, J.M.: Eyediap: A database for the development and evaluation of gaze estimation algorithms from rgb and rgb-d cameras. In: Proceedings of the Symposium on Eye Tracking Research and Applications, pp. 255–258. ACM (2014)
28. Mora, K.A.F., Odobez, J.M.: Person independent 3d gaze estimation from remote rgb-d cameras. In: Image Processing (ICIP), 2013 20th IEEE International Conference on, pp. 2787–2791. IEEE (2013)
29. Nagamatsu, T., Iwamoto, Y., Kamahara, J., Tanaka, N., Yamamoto, M.: Gaze estimation method based on an aspherical model of the cornea: surface of revolution about the optical axis of the eye. In: Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications, pp. 255–258. ACM (2010)
30. Nagamatsu, T., Yamamoto, M., Sato, H.: Mobigaze: development of a gaze interface for handheld mobile devices. In: CHI'10 Extended Abstracts on Human Factors in Computing Systems, pp. 3349–3354. ACM (2010)
31. Naish-Guzman, A., Holden, S.: The generalized fitc approximation. In: Advances in Neural Information Processing Systems, pp. 1057–1064 (2007)
32. Ohno, T., Mukawa, N., Yoshikawa, A.: Freegaze: a gaze tracking system for everyday gaze interaction. In: Proceedings of the 2002 symposium on Eye tracking research & applications, pp. 125–132. ACM (2002)
33. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern Recognit. **29**(1), 51–59 (1996)
34. Perez, A., Cordoba, M.L., Garcia, A., Mendez, R., Munoz, M.L., Pedraza, J.L., Sanchez, F.: A precise eye-gaze detection and tracking system. In: 11th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, pp. 105–108. UNION Agency, Plzen (2003)
35. Porikli, F.: Integral histogram: a fast way to extract histograms in cartesian spaces. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 1, pp. 829–836. IEEE (2005)
36. Rasmussen, C.E., Christopher, K.W.: Gaussian Processes For Machine Learning, vol. 1. MIT Press, Cambridge (2006)
37. Raytchev, B., Yoda, I., Sakaue, K.: Head pose estimation by nonlinear manifold learning. In: Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on, vol. 4, pp. 462–466. IEEE (2004)
38. Schneider, T., Schauerte, B., Stiefelhagen, R.: Manifold alignment for person independent appearance-based gaze estimation. In: Pattern Recognition (ICPR), 2014 22nd International Conference on, pp. 1167–1172. IEEE (2014)
39. Shi, J., et al.: Good features to track. In: Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on, pp. 593–600. IEEE (1994)
40. Shih, S.W., Wu, Y.T., Liu, J.: A calibration-free gaze tracking technique. In: Pattern Recognition, 2000. Proceedings. 15th International Conference on, vol. 4, pp. 201–204. IEEE (2000)
41. Smith, B.A., Yin, Q., Feiner, S.K., Nayar, S.K.: Gaze locking: Passive eye contact detection for human-object interaction. In: Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology, pp. 271–280. ACM (2013)
42. Sugano, Y., Matsushita, Y., Sato, Y.: Learning-by-synthesis for appearance-based 3d gaze estimation. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pp. 1821–1828. IEEE (2014)
43. Tan, K.H., Kriegman, D., Ahuja, N.: Appearance-based eye gaze estimation. In: Applications of Computer Vision, 2002.(WACV 2002). Proceedings. Sixth IEEE Workshop on, pp. 191–195. IEEE (2002)
44. Tomasi, C., Kanade, T.: Detection and tracking of point features. Carnegie Mellon University, Tech. Rep. CMU-CS- 91-132 (1991)
45. Tsang, I.W., Kwok, J.T., Cheung, P.M.: Core vector machines: fast svm training on very large data sets. J. Mach. Learn. Res. **6**, 363–392 (2005)
46. Vatahska, T., Bennewitz, M., Behnke, S.: Feature-based head pose estimation from images. In: Humanoid Robots, 2007 7th IEEE-RAS International Conference on, pp. 330–335. IEEE (2007)
47. Wang, J., Sung, E., Venkateswarlu, R.: Eye gaze estimation from a single image of one eye. In: Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on, pp. 136–143. IEEE (2003)
48. Williams, C.K., Rasmussen, C.E.: Gaussian processes for regression. In: Advances in neural information processing systems, pp. 514–520 (1996)
49. Williams, O., Blake, A., Cipolla, R.: Sparse and semi-supervised visual mapping with the s^3gp. In: Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, vol. 1, pp. 230–237. IEEE (2006)
50. Wollaston, W.H.: On the apparent direction of eyes in a portrait. Philos. Trans. R. Soc. Lond. **114**, 247–256 (1824)
51. Wood, E., Bulling, A.: Eyetab: model-based gaze estimation on unmodified tablet computers. In: Proceedings of the Symposium on Eye Tracking Research and Applications, pp. 207–210. ACM (2014)
52. Ye, Z., Li, Y., Fathi, A., Han, Y., Rozga, A., Abowd, G.D., Rehg, J.M.: Detecting eye contact using wearable eye-tracking glasses. In: Proceedings of the 2012 ACM Conference on Ubiquitous Computing, pp. 699–704. ACM (2012)
53. Yu, S.: Harr feature cart-tree based cascade eye detector homepage. http://yushiqi.cn/research/eyedetection
54. Zhang, X., Sugano, Y., Fritz, M., Bulling, A.: Appearance-based gaze estimation in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4511–4520 (2015)
55. Zhang, Z., Hu, Y., Liu, M., Huang, T.: Head pose estimation in seminar room using multi view face detectors. In: Multimodal Technologies for Perception of Humans, pp. 299–304. Springer (2007)

Qiong Huang received her B.S. degree in Electronic Information Science and Technology from University of Science and Technology of China, Anhui, in 2012 and M.S. in Electrical and Computer Engineering from Rice University, TX, in 2015. Her research interests include computer vision, human gaze analysis and health-related vision applications.

Ashok Veeraraghavan received his Bachelors in Electrical Engineering from the Indian Institute of Technology, Madras, in 2002 and M.S and Ph.D. degrees from the University of Maryland, College Park in 2004 and 2008, respectively. He is currently an Associate Professor of Electrical and Computer Engineering at Rice University, TX, USA. His research interests are broadly in the areas of computational imaging, computer vision and robotics. His thesis received the Doctoral Dissertation Award from the Department of Electrical and Computer Engineering at the University of Maryland.

Ashutosh Sabharwal (S-91-M-99-SM-04) received the B. Tech. degree from the Indian Institute of Technology, New Delhi, India, in 1993 and the M.S. and Ph.D. degrees from The Ohio State University, Columbus, in 1995 and 1999, respectively. He is currently a Professor with the Department of Electrical and Computer Engineering, Rice University, Houston, TX. His research interests include information theory, communication algorithms and experiment-driven design of wireless networks. He received the 1998 Presidential Dissertation Fellowship Award.