Revisiting Data Normalization for Appearance-Based Gaze Estimation

Xucong Zhang
Max Planck Institute for Informatics, Saarland Informatics Campus, Germany
xczhang@mpi-inf.mpg.de

Yusuke Sugano
Graduate School of Information Science and Technology, Osaka University, Japan
sugano@ist.osaka-u.ac.jp

Andreas Bulling
Max Planck Institute for Informatics, Saarland Informatics Campus, Germany
bulling@mpi-inf.mpg.de

ABSTRACT

Appearance-based gaze estimation is promising for unconstrained real-world settings, but the significant variability in head pose and user-camera distance poses significant challenges for training generic gaze estimators. Data normalization was proposed to cancel out this geometric variability by mapping input images and gaze labels to a normalized space. Although used successfully in prior works, the role and importance of data normalization remains unclear. To fill this gap, we study data normalization for the first time using principled evaluations on both simulated and real data. We propose a modification to the current data normalization formulation by removing the scaling factor and show that our new formulation performs significantly better (between 9.5% and 32.7%) in the different evaluation settings. Using images synthesized from a 3D face model, we demonstrate the benefit of data normalization for the efficiency of the model training. Experiments on real-world images confirm the advantages of data normalization in terms of gaze estimation performance.

CCS CONCEPTS

•**Human-centered computing** →*Pointing;* •**Computing methodologies** →*Computer vision;*

KEYWORDS

Eye Tracking; Appearance-based Gaze Estimation; Machine Learning

ACM Reference format:
Xucong Zhang, Yusuke Sugano, and Andreas Bulling. 2018. Revisiting Data Normalization for Appearance-Based Gaze Estimation. In *Proceedings of ACM Symposium on Eye Tracking Research & Applications, Warsaw, Poland, June 2018 (ETRA'18),* 9 pages.
DOI: 10.1145/3204493.3204548

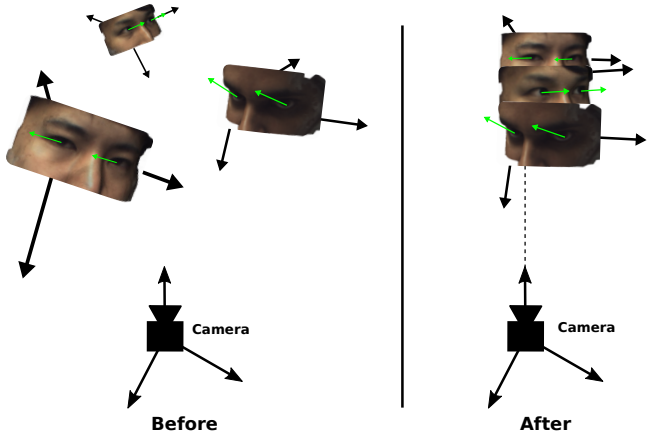

Figure 1: Data normalization, as proposed for appearance-based gaze estimation, cancels out most variations caused by different head poses, by rotating and scaling the images.

1 INTRODUCTION

Driven by advances in deep learning and large-scale training image synthesis, appearance-based gaze estimation methods have recently received increased attention due to their significant potential for real-world applications [Smith et al. 2013; Sugano et al. 2016; Zhang et al. 2018, 2017a,b]. In contrast to their model- and feature-based counterparts [Hansen and Ji 2010; Sesma et al. 2012; Stiefelhagen et al. 1997; Valenti et al. 2012; Venkateswarlu et al. 2003; Wang and Ji 2017; Yamazoe et al. 2008], appearance-based methods aim to directly map eye images to gaze directions, for example obtained using front-facing cameras already integrated into mobile devices [Krafka et al. 2016]. Early methods for appearance-based gaze estimation required a fixed head pose, e.g. enforced using a chin rest [Schneider et al. 2014; Tan et al. 2002; Williams et al. 2006]. While later works allowed for free head rotation [Deng and Zhu 2017; Funes Mora et al. 2014; He et al. 2015; Yu et al. 2016], the distance between user and camera was usually still assumed to be fixed and methods were mainly evaluated in controlled settings. Most recent works focused on the most challenging case, i.e. real-world environments without any constraints regarding head rotation and translation [Krafka et al. 2016; Zhang et al. 2017, 2018].

In principle, given a sufficient amount of training data, the variability caused by unconstrained head pose could be learned from the data. Previous works following this idea consequently focused on significantly increasing the number and diversity of images to train the appearance-based gaze estimator [Krafka et al. 2016; Zhang et al. 2018]. While this approach resulted in significant performance improvements, manual collection and annotation of such large amounts of training data is time-consuming and costly. To

ETRA'18, Warsaw, Poland

978-1-4503-5706-7/18/06…$15.00
DOI: 10.1145/3204493.3204548

reduce the burden of manual data collection, another recent line of work instead proposed to synthesize large numbers of eye images with arbitrary head poses using sophisticated 3D models of the eye region [Wood et al. 2016a,b, 2015]. However, for both of these approaches, covering all possible head poses is nearly impossible. In addition, this approach requires the gaze estimator to deal with a large amount of very similar and mostly redundant data and can result in prolonged training times and more difficult optimization of the loss function.

Data normalization has been proposed to address the aforementioned challenge by reducing the training space and making the training more efficient. This is achieved by preprocessing the training data before it is used as input to the gaze estimator. As shown in Figure 1, the key idea is to normalize the data such that most of the variability caused by different head poses is canceled out. Originally proposed by Sugano et al. [Sugano et al. 2014], this approach has subsequently been used very successfully in other works [Shrivastava et al. 2017a; Zhang et al. 2015, 2017, 2018]. In a nutshell, data normalization first rotates the camera to warp the eye images so that the x-axis of the camera coordinate system is perpendicular to the y-axis of the head coordinate system. Then, the image is scaled so that the (normalized) camera is located at a fixed distance away from the eye center. The final eye images have only 2 degrees of freedom in head pose for all the different data.

Although used successfully in prior works, the importance of rotation and translation/scaling of data normalization remains unclear, and has not yet its impact on the gaze estimation performance been quantified. In this work we aim to fill this gap and, for the first time, explore the importance of data normalization for appearance-based gaze estimation. The specific contributions of this work are two-fold. First, we explain the variability caused by different distances between camera and eye and discuss how data normalization can cancel out some of this variability. Second, we demonstrate the importance of data normalization for appearance-based gaze estimation with extensive experiments on both synthetic and real data. We first perform gaze estimation evaluations on synthesized eye images for different head poses to demonstrate the benefit of applying data normalization. Afterwards, we evaluate within- and cross-dataset settings for gaze estimation and quantify the advantages of data normalization with respect to performance. Third, we propose a modification to the original data normalization formulation and demonstrate that this new formulation yields significant performance improvements for all evaluation settings studied.

2 RELATED WORK

Our work is related to previous works on 1) appearance-based gaze estimation, 2) methods to deal with head pose variability during gaze estimation, and 3) data normalization.

2.1 Appearance-Based Gaze Estimation

Methods for appearance-based gaze estimation aim to directly learn a mapping from eye images to gaze directions. Appearance-based methods are promising because they can be used with low-resolution images, and for long distances between camera and user, given that they do not require explicit eye feature detection. While early works in appearance-based gaze estimation assumed a fixed head pose [Baluja and Pomerleau 1994; Sewell and Komogortsev 2010; Tan et al. 2002], later works specifically focused on allowing for free head rotation [Choi et al. 2013; Funes Mora and Odobez 2012; Lu et al. 2012; Sugano et al. 2008]. These days, free head movement has become a standard requirement for gaze estimation. Consequently, most recent gaze estimation datasets include significant variability in head pose, both with real-world imagery [Deng and Zhu 2017; Funes Mora et al. 2014; Krafka et al. 2016; Smith et al. 2013; Zhang et al. 2018] and synthetic data [Shrivastava et al. 2017b; Sugano et al. 2014; Wood et al. 2015]. Several recent works proposed to learn pose-independent gaze estimators by exploiting large amounts of labeled training data [Shrivastava et al. 2017b; Sugano et al. 2014; Zhang et al. 2017].

2.2 Dealing with Head Pose Variability

Most previous works directly cropped the eye images without any pre-processing, assuming that the model would learn the head pose variability from the training data without additional supervision. Huang et al. used a Haar feature detector to crop the eye images and resized them before gaze estimation training [Huang et al. 2017]. Krafka et al. cropped eyes and face according to landmark detectors and encoded the face size and position as a face grid, which indicated the head position and distance between camera and face [Krafka et al. 2016]. Deng et al. used a head CNN to learn the head pose explicitly from face images to compensate the estimated gaze direction from eye images [Deng and Zhu 2017]. Instead of adding explicit information on face distance or head pose, other works aimed to cancel out variability in head pose using geometric transformations. For example, Lu et al. rotated and translated the camera and then proposed the single-directional pixel flow model to generate the eye image accordingly [Lu et al. 2015]. Mora et al. obtained the frontal face image from a 3D face mesh, which effectively inverted rigid transformations in head pose [Funes Mora and Odobez 2012]. However, all of these methods have additional requirements, such as calibrated eye image [Lu et al. 2015] or a 3D face mesh [Funes Mora and Odobez 2012], and they did not consider the most challenging, but practically also most relevant, task of evaluating gaze estimators across different datasets.

2.3 Data Normalization

Sugano et al. proposed a *data normalization* process to transform eye images and gaze directions into a normalized space to facilitate synthesis of eye images from a 3D face mesh with arbitrary head poses [Sugano et al. 2014]. These synthesized images were then used for gaze estimation. Their basic idea was to rotate and translate the camera to a fixed distance from the eye and to adjust the gaze direction accordingly. Given that images in that normalized space shared the same intrinsic and extrinsic camera parameters, the gaze estimator could be trained and tested in this normalized space. That original data normalization was successfully used in several subsequent works and was key to facilitate cross-dataset evaluations of appearance-based gaze estimation methods [Shrivastava et al. 2017a; Wood et al. 2015; Zhang et al. 2018]. Later works demonstrated that such data normalization could also be used to adapt gaze estimators trained in one setting to new settings, for example to estimate audience attention on public displays [Sugano

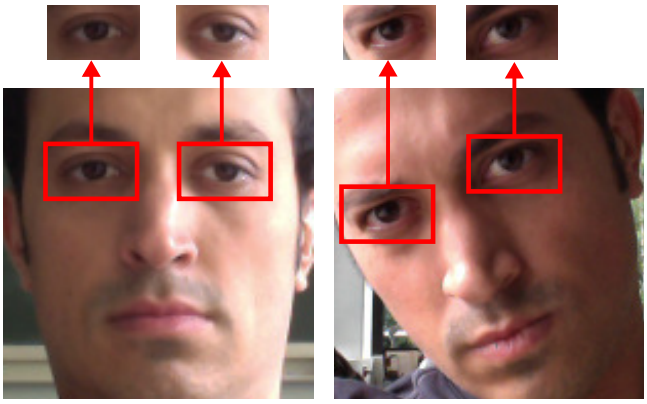

Figure 2: Visualization of head rotation factor. Left: face image and corresponding cropped eye images with nearly non-rotated head pose. Right: face image and corresponding cropped eye images with head pose rotation.

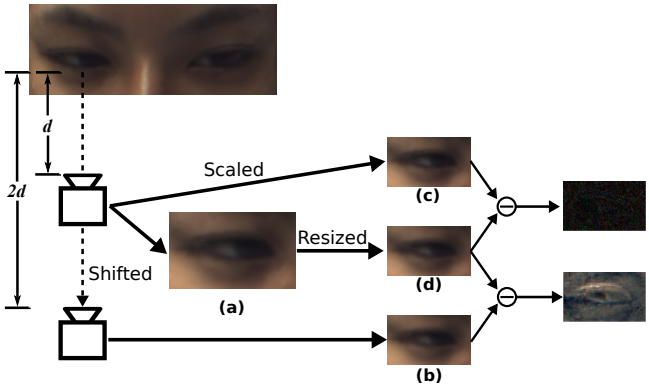

Figure 3: Visualization of distance factor. Eye image (b) is taken at distance d from the camera, and eye image (b) is shifted to distance $2d$ with half the size of (a). Eye images (c) and (d) are the eye images scaled and resized from (a). We calculate the image differences between (b) shifted and (d) resized, and (c) scaled and (d) resized, by subtracting each and normalizing the difference pixel values. Even though it is visually hard to tell, there is an appearance difference between the shifted and resized eye image.

et al. 2016], to detect eye contact independent of the target object type and size, camera position, or user [Müller et al. 2018; Zhang et al. 2017b], or to train person-specific gaze estimators from user interactions across multiple devices [Zhang et al. 2018].

Although data normalization was successfully used in different prior works, it was mainly used to align the training and test data, and its advantage of making the learning-based approach more efficient has not yet been discussed. Also, a principled comparison of gaze estimation performance with and without data normalization is still missing from the current literature.

3 DATA NORMALIZATION

Data normalization aims to align training and test data for learning-based gaze estimation by reducing the variability caused by head rotation and translation. In this section, we first demonstrate the problem setting and discuss why data normalization is needed for canceling out such variability. We describe the detailed process of data normalization presented in prior work [Sugano et al. 2014; Zhang et al. 2018], and point out an issue when handling 2D images. We then introduce our modification on data normalization with a stronger planarity assumption.

3.1 Problem Setting

As discussed earlier, most previous methods on appearance-based gaze estimation assume a frontal head pose, as shown on the left in Figure 2. However, in real-world settings we need to deal with head rotation, as shown on the right in Figure 2. The corresponding eyes are shown above the face image in Figure 2, and the goal of a pose-independent gaze estimator is to estimate 2D gaze positions or 3D gaze directions of eye images no matter how they appear in the original input images.

In addition, precisely speaking, scale/distance of the face also affects the eye appearance. Different distances between camera and eye obviously result in different sizes of eye in the captured images, and the eye appearance itself changes because the eye is not a planar object.

Figure 3 illustrates the effect of distance using a 3D eye region model from the UT Multiview dataset [Sugano et al. 2014]. We capture two eye images at two different distances between eye and camera (Figure 3a and Figure 3b). Closer distance (Figure 3a) naturally results in a larger image resolution, and usually image-based methods resize 2D input images (Figure 3d) so that they have the same resolution size. In this case, although Figure 3a and Figure 3b have the same 3D gaze direction, resized image Figure 3d and further distance image Figure 3b have slightly different image appearances. If we physically scale the 3D space by, e.g., changing the focal length (Figure 3c), the appearance difference between scaled (Figure 3c) and resized images (Figure 3d) is much smaller. This illustrates that image resizing is equivalent to 3D scaling rather than 3D shifting. It is important to precisely discuss the image resizing operation in data normalization.

Pose-independent learning-based methods need to handle these factors causing appearance changes during training processes. However, practically speaking, it is almost impossible to train a gaze estimator with infinite variations of head poses and image resolutions. Therefore, image-based estimation methods require a *normalization* technique to align training and test data and to constrain the input image to have a fixed range of variations. For example, image-based object recognition methods usually crop and resize the input image to a fixed image resolution while assuming that this operation does not affect the object label. The difficulty of data normalization in gaze estimation task is, however, the fact that eye image cropping, rotation, and resizing do affect their corresponding gaze labels. Gaze estimation is inevitably a geometric task, and it is important to properly formulate the normalization operation.

For 3D data, such as UT Multiview [Sugano et al. 2014], EYEDIAP [Funes Mora et al. 2014] and UnityEye [Wood et al. 2015], it is possible to render training and test samples so that they have

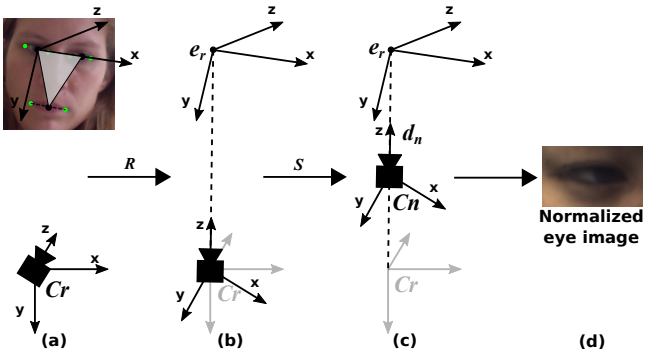

Figure 4: Basic concept of the eye image normalization [Sugano et al. 2014]. (a) Starting from an arbitrary relationship between the head pose coordinate system centered at eye center e_r (top) and the camera coordinate system (bottom); (b) the camera coordinate system is rotated with a rotation matrix R; (c) the world coordinate system is scaled with a scaling matrix S; (d) the normalized eye images should be equivalent to the one captured with this *normalized* camera.

the same camera at the same distance from the eye. However, for captured 2D images, such as MPIIGaze [Zhang et al. 2018] and GazeCapture [Krafka et al. 2016], it is impossible to translate the eye. Nevertheless, we can still perform the approximation to crop the eye image properly.

3.2 Eye Image Normalization

We first summarize the detailed eye image normalization procedure proposed in [Sugano et al. 2014]. The normalization scheme aims at canceling variations in the eye image appearance as much as possible. The key idea is to standardize the translation and rotation between camera and face coordinate system via camera rotation and scaling.

Figure 4 illustrates the basic concept of the eye image normalization. As shown in Figure 4a, the process starts from an arbitrary pose of the target face. The pose is defined as a rotation and translation of the head coordinate system with respect to the camera coordinate system, and the right-handed head coordinate system is defined according to the triangle connecting three midpoints of the eyes and mouth. The x-axis is defined as the line connecting midpoints of the two eyes from right eye to left eye, and the y-axis is defined as perpendicular to the x-axis inside the triangle plane from the eye to the mouth. The z-axis is perpendicular to the triangle and pointing backwards from the face.

To simplify the notation of eye image normalization, we use the midpoint of the right eye as the origin of the head coordinate system, and we denote the translation and rotation from the camera coordinate system to the head coordinate system as $\boldsymbol{e}_r$ and $\boldsymbol{R}_r$.

Given this initial condition, the normalization process transforms the input image so that the normalized image meets three conditions. First, the *normalized* camera looks at the origin of the head coordinate system and the center of the eye is located at the center of the normalized image. Second, the x-axes of the head and camera coordinate systems are on the same plane, i.e., the x-axis of the head coordinate system appears as a horizontal line in the normalized image. Third, the normalized camera is located at a fixed distance d_n from the eye center and the eye always has the same size in the normalized image.

The rotation matrix $\boldsymbol{R}$ to achieve the first and second conditions can be obtained as follows. If we rotate the original camera to meet the first condition, the rotated z-axis of the camera coordinate system $\boldsymbol{z}_c$ has to be $\boldsymbol{e}_r$. To meet the second condition, the rotated y-axis has to be defined as $\boldsymbol{y}_c = \boldsymbol{z}_c \times \boldsymbol{x}_r$. $\boldsymbol{x}_r$ is the x-axis of the head coordinate system, and the y-axis of the rotated camera is defined to be perpendicular to both $\boldsymbol{z}_c$ and $\boldsymbol{x}_r$. Then, the remaining x-axis of the rotated camera is defined as $\boldsymbol{x}_c = \boldsymbol{y}_c \times \boldsymbol{z}_c$. Using these vectors, the rotation matrix can be defined as $\boldsymbol{R} = [\frac{\boldsymbol{x}_c}{\|\boldsymbol{x}_c\|}; \frac{\boldsymbol{y}_c}{\|\boldsymbol{y}_c\|}; \frac{\boldsymbol{z}_c}{\|\boldsymbol{z}_c\|}]$. In addition, the scaling matrix $\boldsymbol{S}$ to meet the third condition can be defined as $\mathrm{diag}(1, 1, \frac{d_n}{\|\boldsymbol{e}_r\|})$. Therefore, the overall transformation matrix is defined as $\boldsymbol{M} = \boldsymbol{SR}$.

In the extreme case where the input is a 3D face mesh, the transformation matrix $\boldsymbol{M}$ can be directly applied to the input mesh and then it appears in the normalized space with a restricted head pose variation. Since the transformation is $\boldsymbol{M}$ defined as rotation and scaling, we can apply a perspective image warping with the transformation matrix $\boldsymbol{W} = \boldsymbol{C}_n\boldsymbol{M}\boldsymbol{C}_r^{-1}$ to achieve the same effect if the input is a 2D face image. $\boldsymbol{C}_r$ is the original camera projection matrix obtained from camera calibration, and $\boldsymbol{C}_n$ is the camera projection matrix defined for the normalized camera.

Sugano et al. [Sugano et al. 2014] introduced this idea to restrict the head pose variation when synthesizing training data for learning-based gaze estimation from 3D face meshes. Since we can assume test data always meets the above three conditions after normalization, it is enough to render training images by placing virtual cameras on a viewing sphere around the eye center with radius d_n and rotating the camera to meet the first and second conditions. This data normalization results in only 2 degrees of freedom, and significantly reduces the training space to be covered via learning-by-synthesis framework.

3.3 Modified Data Normalization

As discussed earlier, it is also important to properly handle the geometric transformation caused by the eye image normalization and apply the same transformation to the gaze direction vector. If the input is training data and associated with a ground-truth gaze direction vector $\boldsymbol{g}_r$, it is necessary to compute the *normalized* gaze vector $\boldsymbol{g}_n$ which is consistent with the normalized eye image.

Assuming 3D data, Sugano et al. [Sugano et al. 2014] originally proposed to apply the same transformation matrix to the gaze vector as $\boldsymbol{g}_n = \boldsymbol{M}\boldsymbol{g}_r$. However, while in the 3D space the same rotation and translation should be applied to the original gaze vector $\boldsymbol{g}_r$, this assumption is not precise enough when dealing with 2D images. Since scaling does not affect the rotation matrix, the head rotation matrix after normalization is computed only with rotation as $\boldsymbol{R}_n = \boldsymbol{R}\boldsymbol{R}_r$. For 2D images, image normalization is achieved via perspective warping as $\boldsymbol{W} = \boldsymbol{C}_n\boldsymbol{M}\boldsymbol{C}_r^{-1}$. This operation implicitly assumes the eye region is a planar object, and if the eye is a planar object, scaling should not change the gaze direction vector.

Based on this discussion, in this work we propose a slightly modified version of the data normalization process for 2D images.

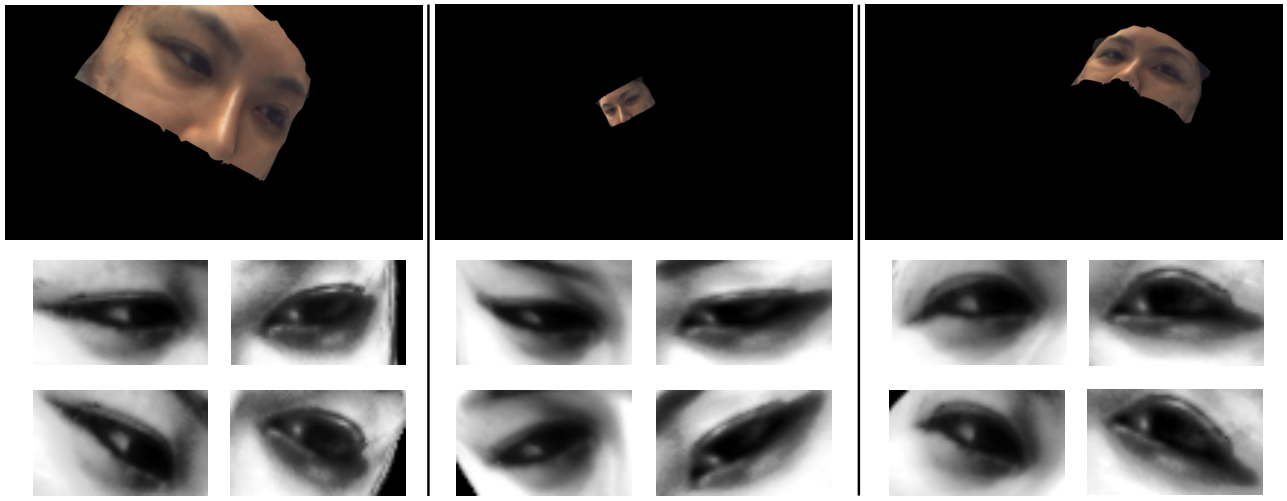

Figure 5: Example of our synthesized 2D images and corresponding eye images from UT Multiview. We first randomly rotated and translated the 3D face mesh in the camera coordinate system to render the 2D image (the top row), and performed the normalization on the captured image to crop the eye image (the middle row), or directly crop the eye images (the bottom row) according to the eye corner landmarks as a naive baseline.

While the formulation of the image normalization and the image transformation matrix $\boldsymbol{W}$ stays exactly the same, different with the original 2D data normalization method, we propose to only rotate the original gaze vector to obtain the normalized gaze vector $\boldsymbol{g}_n = \boldsymbol{R}\boldsymbol{g}_r$. This formulation corresponds to an interpretation of the image transformation matrix $\boldsymbol{W}$ that the scaling $\boldsymbol{S}$ is applied to the camera projection matrix $\boldsymbol{C}_r$, instead of the 3D coordinate system. While this results in the exactly same image warping, it does not affect the physical space in terms of scaling and the gaze vector is only affected with the rotation matrix $\boldsymbol{R}$.

The transformation is also used to project back the estimated gaze vector to the original camera coordinate system. If the gaze vector estimated from the normalized eye image is $\hat{\boldsymbol{g}}_n$, the estimation result in the original camera coordinate system $\hat{\boldsymbol{g}}_r$ is obtained by rotating back $\hat{\boldsymbol{g}}_n$ as $\hat{\boldsymbol{g}}_r = \boldsymbol{R}^{-1}\hat{\boldsymbol{g}}_n$.

4 EXPERIMENTS

In this section, we validate the modified formulation of the data normalization using both synthetic and real image datasets. In all experiments that follow, we used the AlexNet architecture [Krizhevsky et al. 2012] as a basis for our appearance-based gaze estimation network and concatenated the normalized head angle vector and the first fully-connected layer, as done in [Zhang et al. 2018]. The output of the network is a two-dimensional gaze angle vector $\boldsymbol{g}$ as polar angles converted from $\boldsymbol{g}_n$. As loss we used the Euclidean distance between estimated gaze angle vector $\hat{\boldsymbol{g}}$ and ground-truth gaze angle vector $\boldsymbol{g}$. During computing the final gaze estimation error, we first converted $\hat{\boldsymbol{g}}$ and $\boldsymbol{g}$ to $\hat{\boldsymbol{g}}_n$ and $\boldsymbol{g}_n$, and then projected them back to the original camera coordinate system to calculate the differences between direction vectors in degrees. We used the AlexNet pre-trained on the ImageNet dataset [Deng et al. 2009] from the Caffe library [Jia et al. 2014], and fine-tuned the whole network with gaze estimation training data depending on the particular experimental setting (see the respective section below for details). We used the Adam solver [Kingma and Ba 2015] with the two momentum values set to $\beta_1 = 0.9$ and $\beta_2 = 0.95$, as well as the initial learning rate set to 0.00001. For data normalization, we set the focal length for the normalized camera projection matrix and the distance d_n to be compatible with the UT Multiview dataset [Sugano et al. 2014]. The resolution of the normalized eye images was 60×36 pixels.

In this section, we refer to the original data normalization method as *Original*, and the modified data normalization method as *Modified*. We further analyze a naive baseline without any geometric transformation (*None*). For this baseline, we took the center of two eye corners as eye center, 1.5 times of the distance between two eye corners as eye width, and 0.6 times of eye width as eye height to crop the eye image. Last, we resized the eye image to the same 60×36 pixels.

4.1 Evaluation on Synthetic Images

While the main purpose of data normalization is handling large variations in head pose, real-world datasets inevitably have limited head pose variations due to device constraints. To fully evaluate the effect of data normalization on gaze estimation performance, we first use synthetic eye images with controlled head pose variations. We synthesized eye images using 3D face meshes of 50 participants provided by UT Multiview [Sugano et al. 2014] to simulate 2D images that were captured with different head poses. We placed the 3D face mesh at random positions and rotations in the virtual camera coordinate system, and then rendered the image with the camera. The range of these randomizations was [-500 mm, 500 mm] for the x- and y-axes of the 3D face mesh position, [100 mm, 1500 mm] for the z-axis (distance between eye and camera), and [-30°, 30°] for head rotation around the roll, pitch and yaw axes,

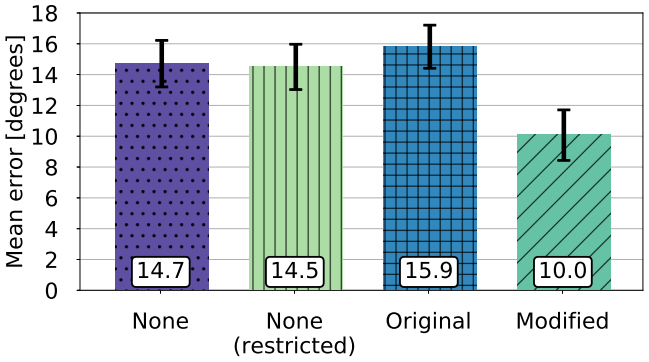

Figure 6: Gaze estimation error in degrees of visual angle for data normalization methods *Original* and *Modified*, and *None* baselines with the gaze estimation network fine-tuned on UT Multiview, and tested on our synthetic samples. Bars show the mean error across participants and error bars indicate standard deviations.

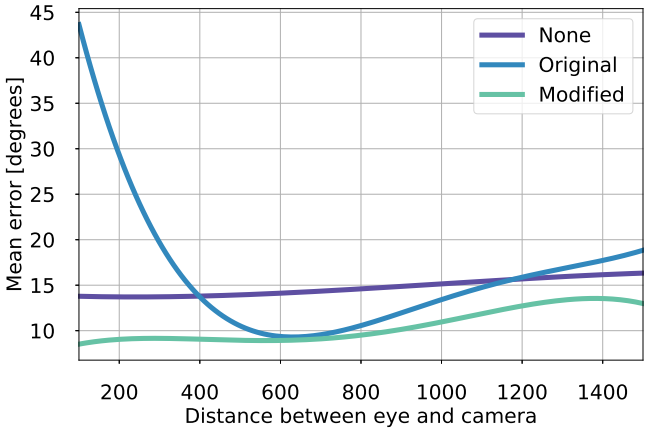

Figure 7: Gaze estimation error for the different normalization methods and different distances between camera and eye. Curves plotted using least squares polynomial fitting.

respectively. Note that we constrained the random position of the 3D face mesh so that the faces always fall inside the camera's field of view. The image resolution was set to 1280×720 pixels. Some examples of the rendered images are shown in the top row of Figure 5. Note that our own synthetic images were treated as 2D images in the following experiments, without access to the original 3D face mesh. The above process is introduced to simulate challenging input images with large head pose variations.

We then performed the data normalization with *Original* or *Modified* methods on the rendered image to crop the eye images. The cropped eye images via 2D data normalization are shown in the middle row of Figure 5, and we also show the cropped eye image from the *None* baseline as the bottom row of Figure 5. Note that the cropped eye images for *Original* and *Modified* are the same as the middle row of Figure 5, and the only difference is whether the gaze direction is scaled or not. UT Multiview contains 160 face meshes with different gaze directions for each 50 participant. Using this approach, we synthesized one 2D image for each face mesh, and flipped the cropped right eye images horizontally and trained them together with the left eye images. This finally resulted in $160 \times 2 = 320$ eye images for each of the 50 participants. Since this *None* baseline cannot take into account the eye position, we also prepared a position-restricted synthetic dataset to train and test a special version (*None (restricted)*) of the *None* baseline. During synthesis, we fixed the x- and y-axes of the 3D face mesh position and set them to zero, and the face center was always located in the image center. This way, only rotation and distance change in this dataset, and the *None (restricted)* baseline takes into account all information related to head pose variation.

4.1.1 Test Data Normalization. To evaluate the effectiveness of the data normalization, we first evaluate the scenario where the training images are synthesized from 3D data under the normalized pose space, and 2D test images are cropped according to the normalization schemes. We fine-tuned the AlexNet model on the synthetic data provided by the original UT Multiview dataset, and tested on our own synthetic samples that were processed with the *Original*, *Modified* or *None* baseline.

We converted the output gaze angle vector from the model $\boldsymbol{g}$ to a gaze direction vector $\boldsymbol{g}_n$, and then projected it back to the original camera coordinate system depending on the normalization method: For *Original*, we computed the gaze direction vector in the original camera space with transformation matrix $\boldsymbol{M}$ as $\boldsymbol{g}_r = \boldsymbol{M}^{-1}\boldsymbol{g}_n$. For the *Modified* method, we computed the gaze direction vector in the original camera space with rotation matrix $\boldsymbol{R}$ as $\boldsymbol{g}_r = \boldsymbol{R}^{-1}\boldsymbol{g}_n$. For the *None* baseline, we directly took the output from the model as the final gaze direction $\boldsymbol{g}_r$ in the original camera space.

The results are shown in Figure 6 with the gaze estimation performances for *None*, *Original* and *Modified*. The bars show the mean gaze estimation error in degrees, and the error bars show the standard deviation across all participants. As can be seen from the figure, the *Modified* method outperforms the other two methods significantly. Since the only difference between *Original* and *Modified* is scaling the gaze direction or not, the better performance achieved by *Modified* over *Original* indicates the scaling on gaze direction actually hurts the performance. This is because the scaling factor is not suitable to apply on gaze direction here, since the eye region in the input image is a planar object. Such a scaling factor even makes the performance worse than the *None* baseline without any data normalization. The *Modified* outperforms over the *None* baseline significantly with 32.0% (from 14.7 degrees to 10.0 degrees), clearly showing the benefits of data normalization for handling the variations caused by head poses. *None (restricted)* achieved slightly better but insignificant performance improvements ($p < 0.01$, paired Wilcoxon signed rank test) over the *None* baseline. This indicates that this naive baseline cannot achieve performance comparable to data normalization even if face positions in the image are restricted.

Figure 7 further shows the gaze estimation error for different distances between camera and eye. To generate a smooth curve, we used least squares polynomial fitting. As can be seen from the figure, the gaze estimation error of the *Modified* method only slightly increases with increasing distance. A similar trend can also

be observed for the *None* baseline. In contrast, the *Original* data normalization method encodes distance information in the gaze direction. This results in an increased gaze estimation error, particularly for small distances. When projecting the gaze direction back to the original camera coordinate system, the gaze direction will be scaled with the inverse scaling matrix $\boldsymbol{S}$. In consequence, the gaze direction is narrowed when the sample has bigger distance than d_n, and the gaze direction is expanded when the sample has smaller distance than d_n. This causes the larger gaze estimation error on the smaller distances. Finally, given that the scaling matrix $\boldsymbol{S}$ for *Original* becomes the identity matrix when the distance between camera and eye is d_n, the gaze estimation error is the same for *Original* and *Modified* at that normalization distance.

4.1.2 Training Data Normalization. In this section, we further evaluate the model trained and tested on the data generated from 2D images. While the model was trained on the data generated directly with the 3D face mesh in the previous evaluation scenario and our synthetic data was used only as test data, in this section we split our synthetic images into training and test data. In this case, the training and test samples were both processed via the *Original*, *Modified* or *None* methods, respectively. We performed a 5-fold cross-person evaluation on the 16,000 synthesized samples from 50 participants.

The results of this evaluation are shown in Figure 8. The bars show the mean gaze estimation error in degrees, and the error bars show the standard deviation across all participants. As can be seen from the figure, in this setting, both data normalization methods achieve better performances than the *None* baseline, suggesting that the data normalization benefits the model training. The *None* baseline performed the worst because the noisy training data with head rotation makes the model training difficult. Restricting the face position does not improve performance, as indicated by the *None (restricted)* baseline. For *Original*, both training and test samples were rotated and also scaled in the same way, which corresponds to mapping the gaze direction into a scaled space. This does not result in large gaze estimation error when projecting the gaze direction back to the original camera coordinate system. However, as we already saw, the *Modified* formulation handles the normalization task more accurately and hence overall performance was still improved.

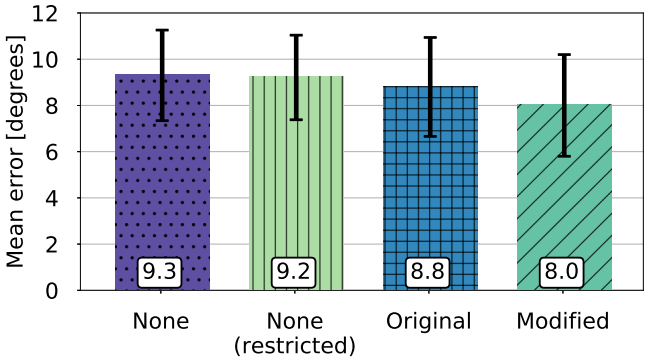

Figure 8: Gaze estimation error in degrees of visual angle for data normalization method *Original* and *Modified*, and *None* baseline with the gaze estimation network fine-tuned and tested on our synthetic samples. Bars show the mean error across participants and error bars indicate standard deviations.

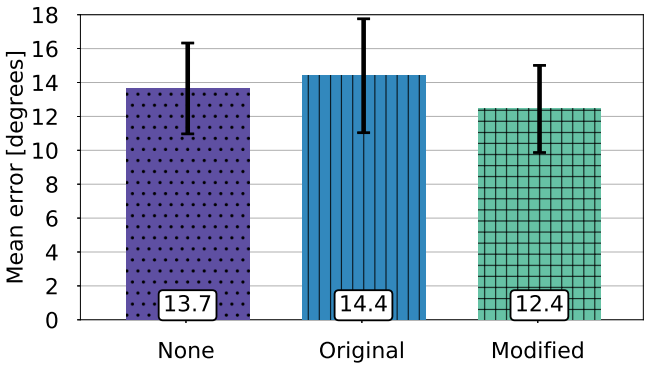

Figure 9: Gaze estimation error in degrees of visual angle for data normalization method *Original* and *Modified*, and *None* baseline with the gaze estimation network fine-tuned on UT Multiview, and tested on MPIIGaze. Bars show the mean error across participants and error bars indicate standard deviations.

4.2 Evaluation on Real Images

We then evaluated the impact of data normalization using real images from the MPIIGaze dataset [Zhang et al. 2018]. As discussed earlier, real images have stronger device constraints, and in terms of head pose, it has smaller variations than the previous case. The MPIIGaze dataset consists of a total of 213,659 images collected on the laptops of 15 participants over the course of several months using an experience sampling approach. Therefore, most of the head poses in the MPIIGaze dataset are restricted to the natural and typical ones in front of a laptop webcam. One important question is whether data normalization contributes to the estimation performance even with a restricted head pose range.

4.2.1 Test Data Normalization. We first performed the simple cross-dataset evaluation, which we trained the model on the UT Multiview dataset and tested on the MPIIGaze dataset. We used the same normalized camera projection matrix, normalized distance ($d_n = 600mm$), and images size (60×36 pixels) as before.

The results are shown in Figure 9. The bars show the mean gaze estimation error in degrees, and the error bars show the standard deviation across all participants. As can be seen from the figure, the ranking in terms of performance is the same as in Figure 6. That is, the *Modified* method achieved the best performance and the *Original* method achieved the worst performance. The *None* baseline has the second-best performance. This analysis confirms that encoding distance information by scaling gaze direction in the *Original* method is not helpful since the eye region is planar in the input 2D image.

The relative improvement achieved by the *Modified* method over the *None* baseline becomes smaller compared to Figure 6 (9.5% vs 32.0%). This is because the head rotation in MPIIGaze data as

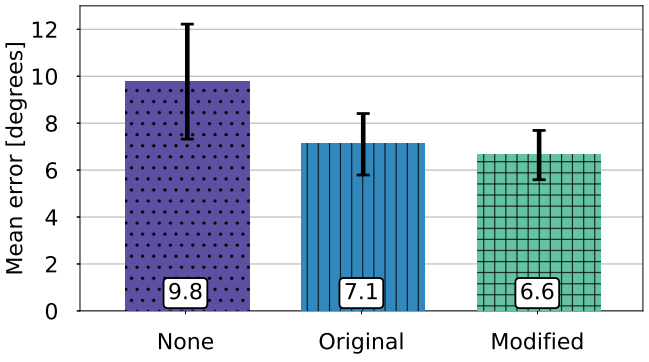

Figure 10: Gaze estimation error in degrees of visual angle for data normalization methods *Original* and *Modified*, and *None* baseline with the gaze estimation network fine-tuned and tested on MPIIGaze. Bars show the mean error across participants and error bars indicate standard deviations.

shown in [Zhang et al. 2018] is much narrower compared to our synthesized samples from UT Multiview.

4.2.2 Training Data Normalization. Last, we repeated the training on 2D images evaluation on MPIIGaze using a leave-one-person-out approach. The training and test sets were both processed via the *Original*, *Modified* or *None* methods, respectively. The results are shown in Figure 10. The bars show the mean gaze estimation error in degrees, and the error bars show the standard deviation across all participants. The figure shows that performance order for the different methods is similar to Figure 8. Both *Original* and *Modified* achieved better performances than the *None* baseline, while *Modified* again achieved the best performance. As such, this analysis confirms that the data normalization can lead to better performance for both synthetic and real data, and that the *Modified* data normalization method can achieve better performance than the *Original* data normalization method.

The relative improvement achieved by the *Modified* method over the *None* baseline when evaluating on synthetic (see Figure 8) and real (see Figure 10) data increased from 16.3% to 32.7% despite the fact that the head rotation range is smaller for real data from MPIIgaze. This is most likely because for the real data, the model has to handle variations that never appeared in synthesized data, such as different illumination conditions. The variability caused by the head rotation becomes crucial during model learning for the *None* baseline since the model has to handle additional variations. This suggests that data normalization is particularly beneficial for the case of training and testing on 2D images, which is the practically most relevant case for appearance-based gaze estimation.

5 CONCLUSION

In this work we modified the data normalization method for appearance-based gaze estimation initially proposed in [Sugano et al. 2014]. We demonstrated the importance of eye image appearance variations caused by different head poses, and provided detailed explanations and discussions on how data normalization can cancel most of these variation to make the model learning more efficient. We showed that data normalization can result in significant performance improvements between 9.5% and 32.7% for different evaluation settings using both synthetic and real image data. These results underline the importance of data normalization for appearance-based methods, particularly in unconstrained real-world settings. As such, we strongly recommend data normalization as the default pre-processing step for appearance-based gaze estimation.

ACKNOWLEDGMENTS

This work was funded, in part, by the Cluster of Excellence on Multimodal Computing and Interaction (MMCI) at Saarland University, Germany, as well as by a JST CREST research grant under Grant No.: JPMJCR14E1, Japan.

REFERENCES

Shumeet Baluja and Dean Pomerleau. 1994. Non-intrusive gaze tracking using artificial neural networks. In *Advances in Neural Inf. Process. Syst.* 753–760.

Jinsoo Choi, Byungtae Ahn, Jaesik Parl, and In So Kweon. 2013. Appearance-based gaze estimation using kinect. In *Proc. IEEE Conf. Ubiquitous Robots and Ambient Intell.* 260–261.

Haoping Deng and Wangjiang Zhu. 2017. Monocular Free-head 3D Gaze Tracking with Deep Learning and Geometry Constraints. In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 3162–3171.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 248–255.

Kenneth Alberto Funes Mora, Florent Monay, and Jean-Marc Odobez. 2014. EYEDIAP: A Database for the Development and Evaluation of Gaze Estimation Algorithms from RGB and RGB-D Cameras. In *Proceedings of the ACM Symposium on Eye Tracking Research and Applications*. ACM. DOI: http://dx.doi.org/10.1145/2578153.2578190

Kenneth Alberto Funes Mora and Jean-Marc Odobez. 2012. Gaze estimation from multimodal Kinect data. In *IEEE Conf. Comput. Vis. Pattern Recognit. Workshop*. 25–30.

Dan Witzner Hansen and Qiang Ji. 2010. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE transactions on pattern analysis and machine intelligence* 32, 3 (2010), 478–500.

Qiuhai He, Xiaopeng Hong, Xiujuan Chai, Jukka Holappa, Guoying Zhao, Xilin Chen, and Matti Pietikäinen. 2015. OMEG: Oulu multi-pose eye gaze dataset. In *Scandinavian Conference on Image Analysis*. Springer, 418–427.

Qiong Huang, Ashok Veeraraghavan, and Ashutosh Sabharwal. 2017. TabletGaze: dataset and analysis for unconstrained appearance-based gaze estimation in mobile tablets. *Machine Vision and Applications* 28, 5-6 (2017), 445–461.

Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. 2014. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 675–678.

Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *The Int. Conf. on Learning Representations* (2015).

Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba. 2016. Eye tracking for everyone. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2176–2184.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

Feng Lu, Yusuke Sugano, Takahiro Okabe, and Yoichi Sato. 2012. Head Pose-free Appearance-based Gaze Sensing via Eye Image Synthesis. In *Proc. IEEE Int. Conf. Pattern Recognit.* 1008–1011.

Feng Lu, Yusuke Sugano, Takahiro Okabe, and Yoichi Sato. 2015. Gaze estimation from eye appearance: a head pose-free method via eye image synthesis. *IEEE Transactions on Image Processing* 24, 11 (2015), 3680–3693.

Philipp Müller, Michael Xuelin Huang, Xucong Zhang, and Andreas Bulling. 2018. Robust Eye Contact Detection in Natural Multi-Person Interactions Using Gaze and Speaking Behaviour. In *Proc. International Symposium on Eye Tracking Research and Applications (ETRA)*. DOI: http://dx.doi.org/10.1145/3204493.3204549

Timo Schneider, Boris Schauerte, and Rainer Stiefelhagen. 2014. Manifold alignment for person independent appearance-based gaze estimation. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*. IEEE, 1167–1172.

Laura Sesma, Arantxa Villanueva, and Rafael Cabeza. 2012. Evaluation of pupil center-eye corner vector for gaze estimation using a web cam. In *Proceedings of the symposium on eye tracking research and applications*. ACM, 217–220.

Weston Sewell and Oleg Komogortsev. 2010. Real-time eye gaze tracking with an unmodified commodity webcam employing a neural network. In *Ext. Abstr. ACM CHI Conf. on Human Factors in Comput. Syst.* 3739–3744.

Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russ Webb. 2017a. Learning from Simulated and Unsupervised Images through Adversarial Training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. 2017b. Learning From Simulated and Unsupervised Images Through Adversarial Training. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Brian A Smith, Qi Yin, Steven K Feiner, and Shree K Nayar. 2013. Gaze locking: passive eye contact detection for human-object interaction. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM, 271–280.

Rainer Stiefelhagen, Jie Yang, and Alex Waibel. 1997. A model-based gaze tracking system. *International Journal on Artificial Intelligence Tools* 6, 02 (1997), 193–209.

Yusuke Sugano, Yasuyuki Matsushita, and Yoichi Sato. 2014. Learning-by-synthesis for appearance-based 3d gaze estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1821–1828.

Yusuke Sugano, Yasuyuki Matsushita, Yoichi Sato, and Hideki Koike. 2008. An incremental learning method for unconstrained gaze estimation. In *Proc. Eur. Conf. Comput. Vis.* 656–667.

Yusuke Sugano, Xucong Zhang, and Andreas Bulling. 2016. AggreGaze: Collective Estimation of Audience Attention on Public Displays. In *Proc. of the ACM Symposium on User Interface Software and Technology (UIST)*. 821–831. DOI: http://dx.doi.org/10.1145/2984511.2984536

Kar-Han Tan, David J Kriegman, and Narendra Ahuja. 2002. Appearance-based eye gaze estimation. In *Applications of Computer Vision, 2002.(WACV 2002). Proceedings. Sixth IEEE Workshop on*. IEEE, 191–195.

Roberto Valenti, Nicu Sebe, and Theo Gevers. 2012. Combining head pose and eye location information for gaze estimation. *IEEE Transactions on Image Processing* 21, 2 (2012), 802–815.

Ronda Venkateswarlu and others. 2003. Eye gaze estimation from a single image of one eye. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 136–143.

Kang Wang and Qiang Ji. 2017. Real Time Eye Gaze Tracking with 3D Deformable Eye-Face Model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1003–1011.

Oliver Williams, Andrew Blake, and Roberto Cipolla. 2006. Sparse and Semi-supervised Visual Mapping with the S^ 3GP. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, Vol. 1. IEEE, 230–237.

Erroll Wood, Tadas Baltrušaitis, Louis-Philippe Morency, Peter Robinson, and Andreas Bulling. 2016a. A 3D Morphable Eye Region Model for Gaze Estimation. In *Proc. European Conference on Computer Vision (ECCV)*. 297–313. DOI: http://dx.doi.org/10.1007/978-3-319-46448-0_18

Erroll Wood, Tadas Baltrusaitis, Louis-Philippe Morency, Peter Robinson, and Andreas Bulling. 2016b. Learning an appearance-based gaze estimator from one million synthesised images. In *Proc. of the 9th ACM International Symposium on Eye Tracking Research & Applications (ETRA 2016)*. 131–138. DOI: http://dx.doi.org/10.1145/2857491.2857492

Erroll Wood, Tadas Baltrusaitis, Xucong Zhang, Yusuke Sugano, Peter Robinson, and Andreas Bulling. 2015. Rendering of eyes for eye-shape registration and gaze estimation. In *Proceedings of the IEEE International Conference on Computer Vision*. 3756–3764.

Hirotake Yamazoe, Akira Utsumi, Tomoko Yonezawa, and Shinji Abe. 2008. Remote gaze estimation with a single camera based on facial-feature tracking without special calibration actions. In *Proceedings of the 2008 symposium on Eye tracking research & applications*. ACM, 245–250.

Pei Yu, Jiahuan Zhou, and Ying Wu. 2016. Learning reconstruction-based remote gaze estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3447–3455.

Xucong Zhang, Michael Xuelin Huang, Yusuke Sugano, and Andreas Bulling. 2018. Training Person-Specific Gaze Estimators from Interactions with Multiple Devices, In Proc. ACM SIGCHI Conference on Human Factors in Computing Systems (CHI). *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)* (2018).

Xiaoyi Zhang, Harish Kulkarni, and Meredith Ringel Morris. 2017a. Smartphone-Based Gaze Gesture Communication for People with Motor Disabilities. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, 2878–2889.

Xucong Zhang, Yusuke Sugano, and Andreas Bulling. 2017b. Everyday Eye Contact Detection Using Unsupervised Gaze Target Discovery. In *Proc. of the ACM Symposium on User Interface Software and Technology (UIST)* (2017-06-26). 193–203.

Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. 2015. Appearance-based Gaze Estimation in the Wild. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 4511–4520. DOI: http://dx.doi.org/10.1109/CVPR.2015.7299081

Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. 2017. It's Written All Over Your Face: Full-Face Appearance-Based Gaze Estimation. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2017-05-18). 2299–2308.

Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. 2018. MPIIGaze: Real-World Dataset and Deep Appearance-Based Gaze Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2018). DOI: http://dx.doi.org/10.1109/TPAMI.2017.2778103