Title

Tracker/Camera Calibration for Accurate Automatic Gaze Annotation of Images and Videos

Authors

Jindal, Swati

Kaur, Harsimran

Manduchi, Roberto

Publication Date

2022-06-08

DOI

10.1145/3517031.3529643

Peer reviewed

Tracker/Camera Calibration for Accurate Automatic Gaze Annotation of Images and Videos

SWATI JINDAL, University of California, Santa Cruz, USA
HARSIMRAN KAUR, University of California, Santa Cruz, USA
ROBERTO MANDUCHI, University of California, Santa Cruz, USA

Modern appearance-based gaze tracking algorithms require vast amounts of training data, with images of a viewer annotated with "ground truth" gaze direction. The standard approach to obtain gaze annotations is to ask subjects to fixate at specific known locations, then use a head model to determine the location of "origin of gaze". We propose using an IR gaze tracker to generate gaze annotations in natural settings that do not require the fixation of target points. This requires prior geometric calibration of the IR gaze tracker with the camera, such that the data produced by the IR tracker can be expressed in the camera's reference frame. This contribution introduces a simple tracker/camera calibration procedure based on the PnP algorithm and demonstrates its use to obtain a full characterization of gaze direction that can be used for ground truth annotation.

CCS Concepts: • **Human-centered computing**;

Additional Key Words and Phrases: gaze tracking, geometric calibration

ACM Reference Format:
Swati Jindal, Harsimran Kaur, and Roberto Manduchi. 2022. Tracker/Camera Calibration for Accurate Automatic Gaze Annotation of Images and Videos. In *2022 Symposium on Eye Tracking Research and Applications (ETRA '22), June 8–11, 2022, Seattle, WA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3517031.3529643

1 INTRODUCTION

There has been substantial recent interest in appearance-based eye gaze tracking technology. The most successful such systems are based on machine learning algorithms that require an extensive amount of annotated image data sets for training. Compared to other applications (e.g., image classification), image annotation for gaze tracking has a fundamental difficulty in that it is hard to precisely measure one's gaze direction from observation of a picture. The standard method for annotating images with gaze direction is to ask subjects to fixate a known location on a screen, typically identified by a specific marker. This procedure directly yields the *gaze point*, in screen coordinates, or, if the camera is geometrically calibrated with the screen, in camera coordinates. Estimation of the actual gaze direction requires identification of the 3-D location of another point along the visual axis (*gaze origin*), which usually implies computing head pose. Several popular data sets have been built this way (see Sec. 2).

This standard procedure, however, has two main drawbacks. First, it inherently generates only sparse samples. Second, the accuracy of gaze direction annotations may be impaired by various factors. For example, it is well known that, during fixation, the gaze is not entirely stable. A study with 5 subjects [Ott et al. 1992] showed horizontal and vertical fluctuations during visual fixation with a standard deviation of approximately 0.1° in both directions, while a later study with 3 subjects [Aytekin et al. 2014] found the average fixational area (defined as the solid angle in which

Manuscript submitted to ACM

gaze remains for 95% of the time during fixation) to be of 1.2 deg^2. Larger deviations were measured for subjects who were myopic [Wahl et al. 2019] or had other forms of visual loss [Leigh et al. 1989]. The errors in pose estimation also contribute to gaze direction errors. For example, for a viewer located at a distance of 50 cm from the camera, a 2 cm depth estimation error results in a 1° gaze reconstruction error when the visual axis is at 30° from the camera's optical axis. In order to enable the acquisition of large, accurately annotated image data sets, some researchers have resorted to synthetic eye images based on carefully designed 3-D models [Wood et al. 2016, 2015]. However, these synthetic images may not represent real-world conditions, as they may fail to model the diversity of morphological characteristics of human faces or the complex photometry of illumination and reflection.

In this work, we propose the use of infrared (IR)-based gaze tracking devices to annotate image data acquired by a camera (such as the webcam in a laptop computer). IR-based gaze tracking is a mature technology [Guestrin and Eizenman 2006; Kar and Corcoran 2017], with a number of commercial devices available for different market sectors (video games [Sundstedt 2012], virtual reality [Piumsomboon et al. 2017], user interface [Khamis et al. 2017], marketing research [Li et al. 2017], optometry [Thomson 2017]). We are considering here desktop-based trackers, which are typically attached to the bottom of a computer screen, and in particular head-pose free trackers, that let the user move their head within a certain range of locations and orientations.

An IR-based tracker can produce measurements (including the visual axis) at a high rate, time-stamped and synchronized with images taken by the camera, providing the desired annotation. Unfortunately, this data is expressed in reference to the tracker frame, not the camera frame. Our main contribution is introducing an easy-to-use procedure to compute the rigid transformation between the two systems. This knowledge allows us to express all geometric-based annotations with respect to the camera frame, making it usable, for example, to train appearance-based gaze tracking algorithms. Our method simply requires a user to directly look at the camera for a few seconds while moving their head to different locations in front of the screen. Note that the camera-tracker calibration is user-independent; that is, it will also work for any other users, provided that the relative geometry of the camera and tracker is not modified (in practice, this means that the tracker should remain rigidly attached to the screen after calibration, or another calibration would be called for). After calibration, large data sets with annotated images can be collected without requiring the subject to fixate specific points on the screen.

In general, we may expect to obtain reliable gaze data annotation by using specialized IR-based tracking devices. For example, the Tobii Pro Nano tracker used in our experiments has a nominal accuracy (average error or bias) of 0.3°, and a nominal precision (RMS error across samples) of 0.1° in optimal conditions (also see [Feit et al. 2017; Zhang et al. 2019] for in-depth performance analysis of a lower quality IR tracker, the Tobii EyeX). However, any residual error in the proposed gaze tracker/camera calibration procedure will contribute to errors in the measured visual axis. We should also point out that our approach can only produce annotations when the user is within the operating range of distances from the tracker (for the Tobii Pro Nano, this is 45–85 cm).

This article is organized as follows. In Sec. 2, we provide the motivation for our work in terms of enabling easy and accurate gaze annotation of images. We then describe our tracker/camera calibration procedure in Sec. 3. Experiments, including comparison with gaze origin computation using a face model, are presented in Sec. 4. Sec. 5 has the conclusions.

2 GROUND-TRUTH GAZE ANNOTATION

2.1 Fixation Method

Several gaze-annotated image data sets have been created and made available for training and testing appearance-based gaze algorithms [Funes Mora et al. 2014; Krafka et al. 2016; Smith et al. 2013; Sugano et al. 2014; Zhang et al. 2020; Zhang et al. 2019]. To create these data sets, participants were asked to move their heads while fixating at specific known locations (gaze points). The visual axis for each eye in each image is estimated by determining a "gaze origin", a generic term for a point on the visual axis located within the eyeball. This is normally obtained using a face model. Note that face models are commonly employed for image normalization (originally introduced by [Sugano et al. 2014], then improved by [Zhang et al. 2018]), with the purpose to cancel out most of the head pose variability. Typically, face detection [Hu and Ramanan 2017] is followed by facial landmarks detection [Deng et al. 2018]. These landmarks are matched with a reference 3-D face model (e.g., the Surrey Face Model [Huber et al. 2016]). For example, [Gross et al. 2010] selects 4 eye corners and 9 nose landmarks to estimate the head pose using PnP [Lepetit et al. 2009]. The 3-D gaze origin is often chosen to be at the midpoint of the segment joining the eye corners. The visual axis is then derived by joining the gaze origin with the gaze point.

2.2 Using an IR Gaze Tracker

An IR tracker computes the *pupillary axis*, that is, the line through the center of corneal curvature and the center of the pupil [Atchison et al. 2000; Nowakowski et al. 2012]. The orientation of the visual axis with respect to the pupillary axis is described by the two *kappa* angles, which are estimated via per-individual calibration that involves fixation on a number of target points on the screen. Note that this calibration procedure is different from the proposed camera/tracker calibration. High-end two-cameras, two-illuminators IR trackers can produce measurements with high accuracy while allowing for free head motion (within a certain range of distance and head orientations).

IR gaze trackers normally provide access through their API (e.g., the Tobii Pro SDK) to two relevant measurements: (1) the *gaze point*, or point of regard, which is the intersection of the visual axis with the screen, expressed in the screen's reference frame (in pixels units); and (2) the *gaze origin*, which is a point on the visual axis contained within the eyeball, expressed in mm in the tracker's reference frame. While it is reasonable to think that the returned location of gaze origin may be at the corneal center of curvature [Guestrin and Eizenman 2006], the Tobii documentation does not give any detail on its actual location, besides it being within the eyeball.

Use of an IR tracker to automatically obtain gaze annotation, rather than rely on discrete fixation targets, may afford data collection in more "natural" situations, such as when reading text, and would enable the collection of larger data sets with ease. Prior work (e.g., the data set described in [Park et al. 2020]) used the gaze point information provided by an IR gaze tracker as a substitute for the location of a fixation pattern. Following the standard procedure described above, the visual axis can then be estimated by joining the location of the gaze point, transformed to the camera's frame reference, with the gaze origin point estimated using a face model. In this work, we propose to use the tracker to provide not only the gaze point but also the gaze origin. It can be expected that the sophisticated procedure by which an IR tracker estimates the location of gaze origin (using corneal reflection from a system using two projectors calibrated with two cameras [Guestrin and Eizenman 2006]) should produce more reliable results than a purely image-based algorithm that relies on a general 3-D model. However, to make use of the gaze origin position estimated by the tracker, it is first necessary to find the relative pose of the camera with respect to the tracker, such that the gaze origin can be expressed in the camera's reference frame. In the next section, we propose a simple calibration procedure that accomplishes that.

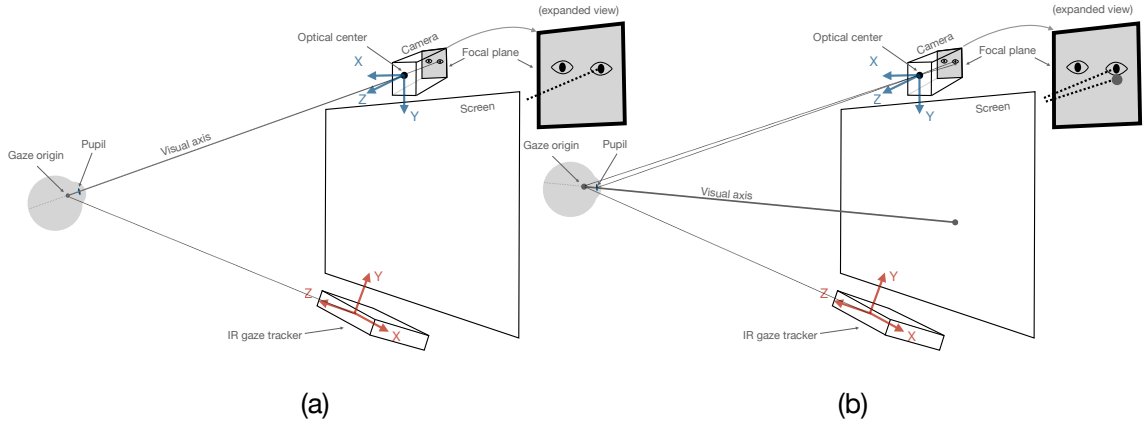

Fig. 1. (a) When the user looks directly at the camera, the visual axis of either eye crosses the camera's optical center, hence that eye's gaze origin projects within the image of the eye's pupil. (b) When the user is looking away from the camera, the gaze origin may project outside of the pupil image.

3 TRACKER/CAMERA CALIBRATION

3.1 The Algorithm

We will assume that the camera and the tracker are rigidly connected in the following. A typical situation is that of a tracker attached at the bottom of a screen (of a laptop or desktop computer), with the camera embedded in the screen. In practice, a gaze tracker could be re-positioned each time it is used (e.g., at the beginning of a session), which would require a new calibration. We also assume that the intrinsic camera parameter matrix K has been estimated, along with the radial distortion parameters.

Our goal is to estimate the relative pose $(R_t^c, \mathbf{T}^c)$ of the IR tracker with respect to the camera, such that a 3-D point $\mathbf{p}^t$ in the tracker's reference frame can be expressed in the camera's frame as $\mathbf{p}^c = R_t^c \mathbf{p}^t + \mathbf{T}^c$. We propose a calibration procedure that uses the 3-D location of the gaze origin for either eye, as estimated by the tracker (hence expressed in the tracker's reference frame). If we can determine the location of the projection of these points onto the camera's focal plane (image) and create multiple pairs (3-D location – 2-D projection) as the user moves their head in different locations, we can use the Perspective-n-Point (PnP) algorithm [Gao et al. 2003] to calibrate camera and tracker. PnP computes the camera's pose from the images of a set of 3-D points in space whose location is known. While PnP is typically applied on a single image of multiple known 3-D points in space, our proposed procedure (with multiple images, each containing two known points in space, namely, the gaze origin of the left and the right eye) is also legitimate.

The problem to be solved, then, is how to find the projection of the gaze origin onto the camera's focal plane. An eye's gaze origin is not directly observable, so there is no simple way to identify it in an image of the user, except for one specific situation: *when the user is looking directly at the camera.* In this case, the image of the gaze origin for either eye can be assumed to be located within the image of that eye's pupil. This is because, in this situation, the visual axis can be assumed to be (approximately) crossing the camera's optical center (as the user is looking at the camera). Hence, all points within the visual axis project onto the same pixel in the camera's focal plane. Given that the visual axis contains the gaze origin and that the visual axis can be expected to go through the pupil, it follows that the image (projection) of the gaze origin should be contained in the pupil's image. Note that the same should *not* be expected

when the user gazes at a point that is away from the camera (e.g., at the opposite end of the screen from where the camera is located; see Fig. 1). For simplicity's sake, and for lack of better information, we will assume that the image of the gaze origin when the user is looking at the camera is located at the center of the pupil image. This enables us to use PnP with gaze origin as a 3-D point, and pupil center as its projection on the image.

Calibration proceeds as follows: the user is asked to move their head to a few different locations within the range covered by the gaze tracker. At each location, the user is asked to look at the camera, and one or more images are acquired. For each image, the pupil's location for each eye is computed. At this point, PnP can be run on the pairs (gaze origin – pupil center) to produce the calibration parameters $(R_t^c, \mathbf{T}^c)$. In the following, we provide some details about our implementation of this proposed calibration mechanism.

3.2 Implementation Details

For our experiments, we attached a small brightly colored paper ring around the camera of the laptop used for data acquisition (MacBook Pro). The colored ring help users to clearly visualize the camera position for effective data collection. Users were asked to position their head at different distances from the camera, and at multiple (9-10) vertical/horizontal locations, within an imaginary cube of approximately $300 \times 300 \times 300$ mm. At each location, users were then asked to look at the colored ring (or directly at the camera) and press a key, after which images and gaze data were collected for 3 seconds (approx. 10 frames).

Automatic detection of the pupil center location is accomplished using the algorithm of [Park et al. 2018][1]. This algorithm computes a set of visual landmarks from the image, which are detected from heatmaps generated by a stacked-hourglass network [Newell et al. 2016] trained on synthetic eye images (UnityEyes [Wood et al. 2016]). We take the pupil center to coincide with the midpoint of the "iris boundary" landmarks. From the 3-D locations of gaze origin (computed by the gaze tracker) and 2-D location of the pupil center detected in each image, we compute the calibration parameters $(R_t^c, \mathbf{T}^c)$ using PnP. We use the `solvePnPRansac` implementation of PnP from OpenCV. This algorithm can take an arbitrary number of points, and is robust to the presence of outliers. Outliers may occur, for example, when the user's gaze unintentionally moves away from the camera.

4 EXPERIMENTS

We conducted a study with five participants (four female), with two main goals: (1) evaluate the accuracy of the proposed calibration algorithm; (2) compare the location of the gaze origin estimated by the IR tracker with that estimated using a face model, and assess the discrepancy between the visual axes computed with the two methods. All participants did not wear eyeglasses, and images were taken in a well-lit environment. Note that, during calibration, one should look for optimal imaging conditions; once calibrated, the system can be used for any users and under any lighting.

Participants sat in front of a 13 inches Apple MacBook Pro, with a Tobii Pro Nano tracker attached (with a magnet) to the bottom of the screen. It is worth mentioning that other configurations, e.g. with the tracker at different locations, are possible. Note that the tracker was repositioned for each new participant, each time requiring a new calibration procedure. Images were acquired by the laptop's camera (1280×720 pixel resolution). These images were timestamped and matched with the closest measurement produced by the tracker. If the tracker did not return a gaze value for either eye (e.g., due to blinking), the associated image was discarded.

[1] https://github.com/swook/GazeML

A standard tracker calibration procedure was first conducted for each participant, using the Tobii Pro Eye Tracker Manager utility with a 9-point fixation pattern. The calibration was then evaluated by asking the participant to again fixate on the markers of the same 9-point fixation marker, and measuring the average angular error (we used the validation code provided by Tobii[2]). We verified that, for each participant, the average angular error for both eyes was less than 1°.

After IR tracker calibration, each participant performed the camera/tracker calibration procedure detailed in the prior section. In addition, we conducted a second data collection exercise which is mainly used for the final evaluation of the quality of tracker-camera calibration. Participants were asked to look at a marker appearing in turn at 9 different locations of a regular calibration pattern. The purpose of this was to obtain images and associated gaze data for a representative range of gaze directions. The actual location of the marker on the screen (which is critical for fixation-type calibration procedures) was irrelevant for this study. For this data acquisition, participants were asked to find a comfortable position to fixate the points of the pattern. The distance between the participants' head and camera varied between 55 cm and 70 cm, and the measured gaze varied within a range of approximately 25° (pitch) by 30° (yaw). Approximately 150 images were acquired for each participant during this second data collection.

4.1 Calibration Accuracy Evaluation

We considered two metrics for evaluating the accuracy of tracker/camera calibration. The first metric is *reprojection error* e_R: each gaze origin location for either eye computed by the tracker for the images used in calibration (with the user looking at the camera) was transformed to the camera's reference frame using the parameters estimated by PnP, then projected onto the cameras' focal plane through the intrinsic parameter matrix K. If PnP was successful, the reprojection error (distance between this projected point and the pupil center location for that eye) should be small. To evaluate e_R, we consider the same data points used to compute $(R_t^c, \mathbf{T}^c)$, with users looking at camera. For each participant, we computed the reprojection error (in pixels) for all gaze origin points within the inlier set determined by the PnP algorithm. Note that the proportion of inliers across participants varied between 41% and 70%. The average reprojection error per participant is shown in Tab. 1. Note that this is consistently less than 1 pixel. Sample images from the inlier set are shown in Fig. 2, showing good localization of the gaze origin projection within the corresponding eye's pupil image.

The second evaluation metric considered, *pupil inconsistency* e_P, was measured on the second data set (with participants looking at different points on the screen, away from the camera). For this evaluation, we first computed the screen/camera calibration using the algorithm of [Rodrigues et al. 2010]. Then, for each measurement, we transformed both gaze point and gaze origin (as produced by the tracker) into the camera's reference frame, using the obtained parameters $(R_t^c, \mathbf{T}^c)$ from PnP. The line joining these two points is the estimated visual axis, which is then projected onto the camera's focal plane using the intrinsic camera matrix K. Since the visual axis can be assumed to go through the eye's pupil, its projection should cross the pupil image. Accordingly, we define a measure of inconsistency as the distance between the projected visual axis and the pupil image. We measure this quantity as follows. First, we determine the radius r of the pupil image (assumed circular for simplicity's sake) by computing the foreshortening of the actual pupil radius R, which is a measurement provided by the Tobii Pro SDK: $r = f \cdot R/Z$, where f is the camera's focal length and Z is the distance between the pupil and the camera. We then measure the distance d between the projected visual axis and the pupil center (where the latter is computed using the algorithm of [Park et al. 2018]), and define:

[2]https://github.com/tobiipro/prosdk-addons-matlab

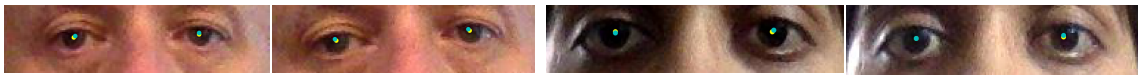

Fig. 2. Sample images collected for tracker/camera calibration. Note that the participants are looking at the camera, with their head moving in different locations between images. The pupil center location is shown as a yellow dot, while the projection of the gaze origin (as computed by the IR tracker) is shown colored in aqua. These images belong to the set of inliers as determined by the PnP algorithm.

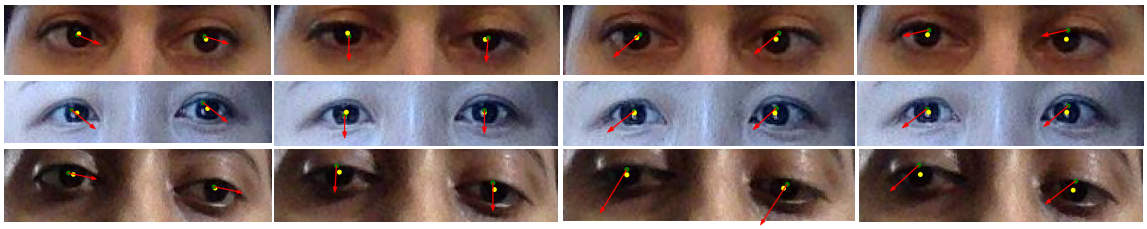

Fig. 3. Sample images of participants looking at different locations on the screen. The pupil center is shown as a yellow dot, while the projection of the gaze origin (as computed by the IR tracker) is shown colored in green. The red arrow shows the projection of a 40 mm long segment, starting from the gaze origin and aligned along the visual axis. Note that in the first three columns, the projection of the visual axis crosses the pupil image ($e_P = 0$). For the images in the fourth column, $e_P > 0$.

$e_P = \max(0, d - r)$. Note that $e_P = 0$ when the visual axis projection crosses the pupil image. The values e_P, averaged over all images in the second data set for each participant, are shown in Tab. 1. Fig. 3 shows examples with $e_P = 0$ (first three columns) and with $e_P > 0$ (fourth column).

Participant	e_R (pixels)		e_P (pixels)		e_G (mm)		e_V (degs)	
	Left	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**
P1	0.62	0.59	0.40	0.36	48.9	45.4	1.91	1.86
P2	0.57	0.69	0.49	0.16	82.2	81.2	1.81	2.01
P3	0.58	0.54	0.97	0.91	62.0	60.9	1.64	1.53
P4	0.72	0.75	0.42	0.94	77.6	76.0	2.68	2.62
P5	0.56	0.70	0.84	0.45	49.3	45.6	1.28	1.63

Table 1. Experimental results for the left and right eyes of our participants. e_R: reprojection error. e_P: pupil inconsistency error. e_G: distance between gaze point location computed with a 3-D face model [Huber et al. 2016] and with the IR tracker. e_V: the angular difference between the associated visual axes. Note that e_R values were computed on the images used for tracker/camera calibration (with the participants looking at the camera), while the other measurements are for the second data set, with the participants looking at different locations on the screen.

4.2 Gaze Origin Computation: IR Tracker vs. Face Model

The proposed tracker/camera calibration enables the use of IR trackers to accurately measure the gaze origin (expressed in the camera's reference frame), which can be used to compute the visual axis as the line joining the gaze origin with the gaze point. It is interesting to compare the location of the gaze origin from the IR tracker with that computed using a face model. We used the 3-D face model from [Huber et al. 2016], as described in Sec. 2.1, to estimate the gaze origin for all images in our second data set.

Tab. 1 shows the average distance e_G between the gaze origin locations produced by the two procedures. The large discrepancy (up to 82 mm of distance) can be explained in large part by depth (Z) errors in the data from the face model, possibly due to imperfect fit of the 3-D model. The sample images of Fig. 4 show, for each eye, the projection of gaze origin, estimated by the two methods, along with the detected eye corners, which are used to compute the gaze origin using the face model (Sec. 2.1.) Note that, at least for these examples, the gaze origin from the face model (red dots) is clearly incorrect (as it appears to be below the pupil, which would indicate an upward gaze, while in these images, the participants were looking at a point below the camera). Fig. 4 also shows the projection of the visual axes, obtained by joining the two different gaze origins with the same gaze point (from the IR tracker). The average angle between these two estimated visual axes, e_V, is shown for each participant in Tab. 1.

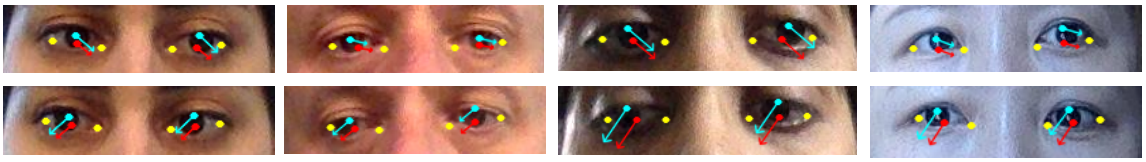

Fig. 4. Sample images of participants looking at different locations on the screen, with a 40 mm segment of visual axis shown starting from the gaze origin, computed using a 3-D face model (shown with red color) and from the IR tracker (shown in aqua color), and joining the same gaze point (as computed by the IR tracker). The 2-D image projections of the eye corners are shown in yellow color.

5 CONCLUSIONS

IR gaze trackers can be very useful for the collection of large gaze-annotated image data sets. Unlike traditional modalities requiring fixation of specific locations, IR trackers make it possible to measure gaze in dynamic settings, e.g., while reading text on the screen. In order to leverage the 3-D data produced by the IR tracker (and not just the gaze point on the screen), it is necessary to first find the relative pose of the tracker with respect to the camera. We have proposed a simple calibration procedure that asks the user to simply look at the camera from different head locations.

Our tracker/camera calibration enables determination of the visual axis in the camera's reference frame, obtained by joining the "gaze origin" produced by the IR tracker with the gaze point on the screen, also computed by the tracker. In our experiments, we compare this quantity with that obtained by joining the gaze point with a different gaze origin location, computed through a 3-D face model (which is the standard procedure to obtain gaze direction from fixation). Our results show that the average angular difference between these two axes can reach values as large 2°, suggesting that using a 3-D face model to estimate the gaze origin may introduce non-negligible errors.

Of course, there are clear limitations to the use of IR trackers for gaze annotation of images. The range of head locations and orientations from which gaze can be computed accurately is constrained. While appropriate for interaction with a laptop or desktop computer, an IR tracker may not be used for applications that call for larger viewing distances or angles (see also [Zhang et al. 2019]). In addition, tracking accuracy is critical if this is to be used for ground-truth measurements, which means that only high quality (and thus expensive) models (such as the Tobii Pro Nano used in this study) should be used for this purpose.

ACKNOWLEDGMENTS

Research reported in this publication was supported by the National Eye Institute of the National Institutes of Health under award number R01EY030952-01A1. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

REFERENCES

David A Atchison, George Smith, and George Smith. 2000. *Optics of the human eye*. Vol. 2. Butterworth-Heinemann Oxford.

Murat Aytekin, Jonathan D Victor, and Michele Rucci. 2014. The visual input to the retina during natural head-free fixation. *Journal of Neuroscience* 34, 38 (2014), 12701–12715.

Jiankang Deng, Yuxiang Zhou, Shiyang Cheng, and Stefanos Zaferiou. 2018. Cascade multi-view hourglass model for robust 3d face alignment. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*. IEEE, 399–403.

Anna Maria Feit, Shane Williams, Arturo Toledo, Ann Paradiso, Harish Kulkarni, Shaun Kane, and Meredith Ringel Morris. 2017. Toward everyday gaze input: Accuracy and precision of eye tracking and implications for design. In *Proceedings of the 2017 Chi conference on human factors in computing systems*. 1118–1130.

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* (Safety Harbor, Florida, United States of America). ACM. https://doi.org/10.1145/2578153.2578190

Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang, and Hang-Fei Cheng. 2003. Complete solution classification for the perspective-three-point problem. *IEEE transactions on pattern analysis and machine intelligence* 25, 8 (2003), 930–943.

Ralph Gross, Iain Matthews, Jeffrey Cohn, Takeo Kanade, and Simon Baker. 2010. Multi-pie. *Image and vision computing* 28, 5 (2010), 807–813.

Elias Daniel Guestrin and Moshe Eizenman. 2006. General theory of remote gaze estimation using the pupil center and corneal reflections. *IEEE Transactions on biomedical engineering* 53, 6 (2006), 1124–1133.

Peiyun Hu and Deva Ramanan. 2017. Finding tiny faces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 951–959.

Patrik Huber, Guosheng Hu, Rafael Tena, Pouria Mortazavian, P Koppen, William J Christmas, Matthias Ratsch, and Josef Kittler. 2016. A multiresolution 3d morphable face model and fitting framework. In *Proceedings of the 11th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*.

A. Kar and P. Corcoran. 2017. A Review and Analysis of Eye-Gaze Estimation Systems, Algorithms and Performance Evaluation Methods in Consumer Platforms. *IEEE Access* 5 (2017), 16495–16519.

Mohamed Khamis, Axel Hoesl, Alexander Klimczak, Martin Reiss, Florian Alt, and Andreas Bulling. 2017. Eyescout: Active eye tracking for position and movement independent gaze interaction with large public displays. In *Proceedings of the 30th annual ACM symposium on user interface software and technology*. 155–166.

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.

RJ Leigh, SE Thurston, RL Tomsak, GE Grossman, and DJ Lanska. 1989. Effect of monocular visual loss upon stability of gaze. *Investigative ophthalmology & visual science* 30, 2 (1989), 288–292.

Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. 2009. Epnp: An accurate o (n) solution to the pnp problem. *International journal of computer vision* 81, 2 (2009), 155.

Yixuan Li, Pingmei Xu, Dmitry Lagun, and Vidhya Navalpakkam. 2017. Towards measuring and inferring user interest from gaze. In *Proceedings of the 26th International Conference on World Wide Web Companion*. 525–533.

Alejandro Newell, Kaiyu Yang, and Jia Deng. 2016. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*. Springer, 483–499.

Maciej Nowakowski, Matthew Sheehan, Daniel Neal, and Alexander V Goncharov. 2012. Investigation of the isoplanatic patch and wavefront aberration along the pupillary axis compared to the line of sight in the eye. *Biomedical optics express* 3, 2 (2012), 240–258.

Dietmar Ott, Scott H Seidman, and R John Leigh. 1992. The stability of human eye orientation during visual fixation. *Neuroscience letters* 142, 2 (1992), 183–186.

Seonwook Park, Emre Aksan, Xucong Zhang, and Otmar Hilliges. 2020. Towards End-to-end Video-based Eye-Tracking. In *European Conference on Computer Vision*. Springer, 747–763.

Seonwook Park, Xucong Zhang, Andreas Bulling, and Otmar Hilliges. 2018. Learning to find eye region landmarks for remote gaze estimation in unconstrained settings. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*. 1–10.

Thammathip Piumsomboon, Gun Lee, Robert W Lindeman, and Mark Billinghurst. 2017. Exploring natural eye-gaze-based interaction for immersive virtual reality. In *2017 IEEE Symposium on 3D User Interfaces (3DUI)*. IEEE, 36–39.

Rui Rodrigues, Joao P Barreto, and Urbano Nunes. 2010. Camera pose estimation using images of planar mirror reflections. In *European Conference on Computer Vision*. Springer, 382–395.

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*. 271–280.

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.

Veronica Sundstedt. 2012. Gazing at games: An introduction to eye tracking control. *Synthesis Lectures on Computer Graphics and Animation* 5, 1 (2012), 1–113.

David Thomson. 2017. Eye tracking and its clinical application in optometry. *Optician* 2017, 6 (2017), 6045–1.

Siegfried Wahl, Denitsa Dragneva, and Katharina Rifai. 2019. The limits of fixation—Keeping the ametropic eye on target. *Journal of vision* 19, 13 (2019), 8–8.

Erroll Wood, Tadas Baltrušaitis, Louis-Philippe Morency, Peter Robinson, and Andreas Bulling. 2016. Learning an appearance-based gaze estimator from one million synthesised images. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*. 131–138.

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.

Xucong Zhang, Seonwook Park, Thabo Beeler, Derek Bradley, Siyu Tang, and Otmar Hilliges. 2020. ETH-XGaze: A Large Scale Dataset for Gaze Estimation under Extreme Head Pose and Gaze Variation. *arXiv preprint arXiv:2007.15837* (2020).

Xucong Zhang, Yusuke Sugano, and Andreas Bulling. 2018. Revisiting data normalization for appearance-based gaze estimation. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*. 1–9.

Xucong Zhang, Yusuke Sugano, and Andreas Bulling. 2019. Evaluation of appearance-based methods and implications for gaze-based applications. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13.

X. Zhang, Y. Sugano, M. Fritz, and A. Bulling. 2019. MPIIGaze: Real-World Dataset and Deep Appearance-Based Gaze Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41, 1 (2019), 162–175.