Task-embedded online eye-tracker calibration for improving robustness to head motion

Jimin Pi
Hong Kong University of Science and Technology
Hong Kong
jpi@connect.ust.hk

Bertram E. Shi
Hong Kong University of Science and Technology
Hong Kong
eebert@ust.hk

ABSTRACT

Remote eye trackers are widely used for screen-based interactions. They are less intrusive than head mounted eye trackers, but are generally quite sensitive to head movement. This leads to the requirement for frequent recalibration, especially in applications requiring accurate eye tracking. We propose here an online calibration method to compensate for head movements if estimates of the gaze targets are available. For example, in dwell-time based gaze typing it is reasonable to assume that for correct selections, the user's gaze target during the dwell-time was at the key center. We use this assumption to derive an eye-position dependent linear transformation matrix for correcting the measured gaze. Our experiments show that the proposed method significantly reduces errors over a large range of head movements.

CCS CONCEPTS

• **Human-centered computing** → **Human computer interaction (HCI)**; **Interaction devices**;

KEYWORDS

Eye tracking, Remote eye tracker, Online calibration, Head movement, Human-Computer Interaction

ACM Reference Format:
Jimin Pi and Bertram E. Shi. 2019. Task-embedded online eye-tracker calibration for improving robustness to head motion. In *2019 Symposium on Eye Tracking Research and Applications (ETRA '19), June 25–28, 2019, Denver , CO, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3314111.3319845

1 INTRODUCTION

Gaze is an important signal for communication and interaction. For people who lose their arms or who have little or no control of their muscles, gaze is one of the main channels for them to communicate and interact with the world. With the development of technology, there have been many gaze-based interaction applications. As reviewed in [Kar and Corcoran 2017], the main applications can be broadly classified into (i) desktop computers [Corcoran et al. 2012] [Pi and Shi 2017] [Chen and Shi 2018] [Dong et al. 2015] (ii) TV panels [Lee et al. 2010] (iii) head mounted displays [Ryan et al. 2008] (iv) automotive setups [Ji and Yang 2002] and (v) hand-held devices [Nagamatsu et al. 2010]. Applications based on desktop platforms include using gaze for computer control, computer communication and text entry, playing games etc. For most screen-based applications, the user's gaze is usually located on a 2D computer screen through remote eye trackers.

One of the key challenges in using gaze for human computer interaction is limited eye tracking accuracy. For example, for gaze typing where all keys are presented on a virtual keyboard on the screen, the size of each key is usually limited. Small errors may cause incorrect selection of neighboring keys to the desired key. With the continuing development of eye tracking technology, we now have many choices of commercial eye trackers. Although manufactures usually report an accuracy $\leq 0.5°$, the specifications are usually achieved under optimal conditions, e.g. constant room illumination and limited head movements. [Hessels et al. 2015] and [Niehorster et al. 2018] compared the performance of several popular commercially available remote eye trackers' performance in non-optimal conditions. They evaluated the performance when the subject looked away from and back to the screen, when one of the eyes was occluded, and when the head orientation changed in roll, pitch and yaw. Eye-tracking quality deteriorates when the participant is unrestrained and assumes a non-optimal pose in front of the eye tracker. This is consistent with our own experience. For tasks calling for high eye tracking accuracy, users need to do recalibration regularly as the accuracy deteriorates over time.

Remote eye tracking methods can be classified into two main groups [Hansen and Ji 2010]: appearance-based methods and model-based methods. Appearance-based methods directly map image features to gaze points. The systems usually only require an ordinary camera. However, the estimation accuracy is usually not sufficient for applications such as computer control or text entry. Model-based methods mostly estimate gaze direction using 3D geometric eye models. One common methodology is the Pupil Center Corneal Reflection (PCCR) method. Near-infrared light emitting diodes are used to produce glints on the eye cornea surface. Gaze direction, or the point of regard (PoR), is estimated from the relative movement between the pupil center and glint positions in the eye region of the captured image. The intersection of the screen and gaze direction is the estimated gaze position on the screen. Products using 3D eye model-based methods claim to be robust to head movement. The performance will deteriorate if the subject moves too fast, turns his head to the side or moves to the edges of the trackbox [Tobiipro 2019]. However, in practice these products often only work within limited range of head movements. Performance degrades with the natural movement of the user's head [Jung et al. 2016]. When the

ETRA '19, June 25–28, 2019, Denver , CO, USA

ACM ISBN 978-1-4503-6709-7/19/06…$15.00
https://doi.org/10.1145/3314111.3319845

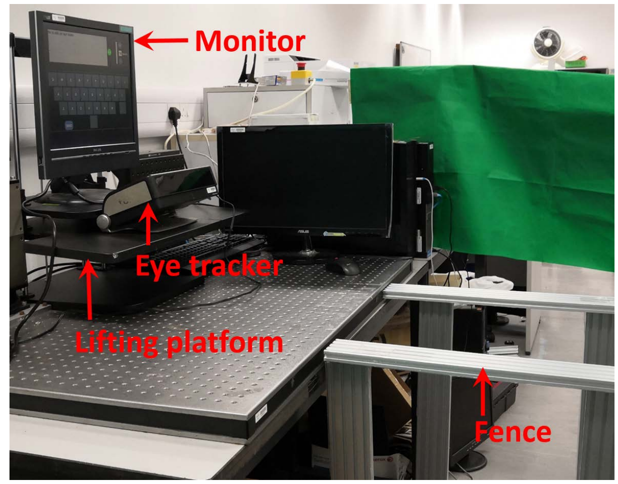
user moves with respect to the tracker-camera axis, changes in left and right eye make it difficult to estimate the corneal reflections and pupil center accurately. These effects limit their application when precision and accuracy are required. Using a pan-tilt camera can ameliorate the problem [Ohno and Mukawa 2004]. However, very few remote eye trackers use this technology. A notable exception is the Eyefollower [EyeFollower 2019].

There also exists another remote eye tracking method called the cross-ratio(CR)-based method, which is used in a small proportion of trackers. Four LEDs on four corners of a computer screen are used to produce glints on the surface of the cornea. From the glint positions, the pupil and the size of the monitor screen, gaze location is estimated by exploiting the cross-ratio of the four points in projective space. These methods make some simplifying assumptions that lead them to be sensitive to head movements. The pupil center and the glints are assumed to lie on the same plane. The PoR is assumed to lie on the optical axis, rather than the visual axis, which deviates from the optical axis by an angle *kappa*. Extensions to the CR methods have been successful in reducing angular error. Yoo and Chung addressed the non-coplanarity by adding a fifth light source and introducing a scale factor for each glint [Yoo and Chung 2005]. Coutinho compensated for the angle *kappa* by introducing a displacement vector [Coutinho and Morimoto 2013]. Hansen proposed a normalized homography mapping to improve the robustness against perspective distortions [Hansen et al. 2010].

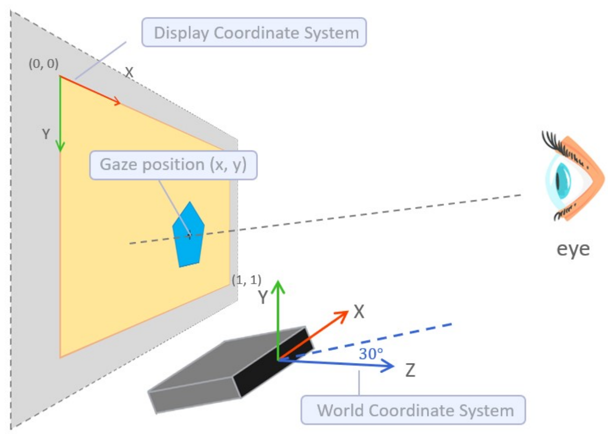
Although more intrusive than a remote eye tracker, a head-mounted eye tracker is a direct solution to handle large head movements. For desktop computer-based interactions, a real-time transformation matrix is needed to transform the gaze from head-centric to screen-centric coordinates. Wang et al. used a simultaneous localization and mapping (SLAM) algorithm to estimate the transformation matrix to a static world coordinate system online [Wang et al. 2018]. However, the accuracy reported (2.8°at a distance of 65cm) is not good enough for precise desktop interactions.

To improve robustness to head movement, we propose here a method for online calibration of model-based remote eye trackers. The system assumes an initial calibration performed at a single calibration location in front of the eye tracker (as most manufacturers require). As time goes on, the algorithm collects triples of (eye position, raw gaze estimate, gaze target), and uses them to construct an eye position dependent mapping from raw gaze estimates to corrected gaze estimates. The assumption that gaze targets are available is not overly restrictive as these can be estimated in a number of task-embedded ways. For example, in dwell-time based gaze selection, it is reasonable to assume that the user's intended gaze target during the dwell-time is at the button center of a correct selection.

Some past work has investigated online calibration during gaze-based human-computer interaction. Several authors have proposed methods for estimating a head and gaze position independent offset between either the estimated and actual gaze points in pixels [Hornof and Halverson 2002; Zhang and Hornof 2011] or between the optical axis and the visual axis (*kappa*) [Chen and Ji 2015]. In the first approach, the offset was estimated from the histogram of disparities between the gaze points and their nearest object points. The second approach merged image dependent probability distributions over gaze with an estimated probability distribution over



(a)

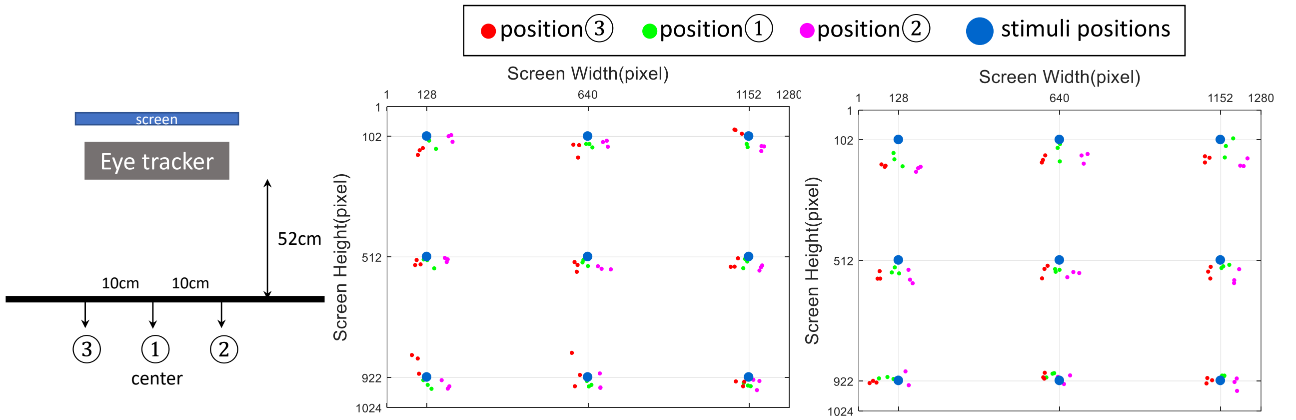


(b)

Figure 1: (a)The experiment setup. (b) The World Coordinate System for eye positions and the Display Coordinate System for gaze positions.

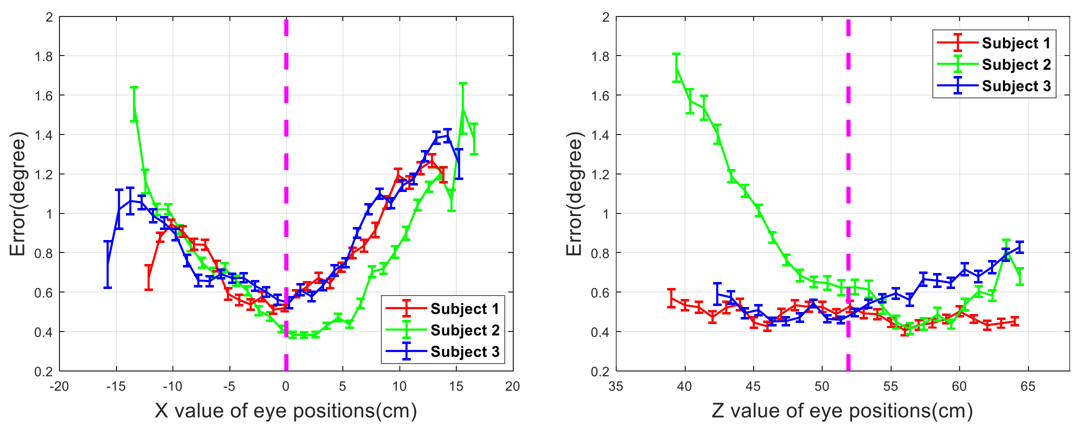
kappa updated online based on past images and measurements of the optical axis. Our work differs primarily in that it considers a head-position dependent and affine correction. [Sugano et al. 2015] updated the parameters of their appearance-based gaze estimator by augmenting the training set with image/interaction target pairs collected from mouse clicks. [Huang et al. 2016] showed that this could be further improved by a better method for identifying image/interaction target pairs. Our work differs primarily in that we do not require explicit knowledge of the gaze estimation model, and thus should be more widely applicable to existing and commercially-available eye tracking systems.

In contrast, we show here that the degradation of gaze estimation due to head movement in a PCCR-based remote eye tracker is actually position dependent and stable over time (Section 2). This suggests that errors caused by changes in position can be compensated. However, the degradation is subject dependent, and



(a) (b) (c)

Figure 2: Position-dependent sensitivity to head movement. (a) Experiment positions(top view). (b)(c)Each plot shows the results from one subject.



(a) (b)

Figure 3: Subject-dependent sensitivity to head movement. Error bars indicate the standard error, the vertical pink dashed line indicates the initial position. (a) Moving left and right. (b) Moving forward and backward.

therefore cannot be compensated in advance. Rather, the compensation must be estimated for each subject individually. To avoid a lengthy calibration process, we propose here to collect the data required to compensate the gaze estimates on-line as the subject uses the eye tracker. As we make no assumptions about how the raw gaze estimates are generated, but rather directly map raw gaze to corrected gaze, we believe this algorithm is widely applicable. We expect that our approach can be applied directly to a wide range of commercially available remote eye trackers, as long as the raw gaze estimates degrade systematically as a function of position.

2 GAZE ESTIMATE DEPENDENCY ON HEAD MOVEMENT

Although it is generally recognized that remote eye trackers' performance degrades over time, the reason for this is not clear. We report here on two experiments. The first demonstrates that eye gaze estimates degrade systematically with position, but that the degradation is stable over time. The second demonstrates that the degradation is subject dependent.

Figure 1(a) shows the experimental setup. A Tobii X60 remote eye tracker placed under the screen with tilt angle of 30°provides gaze estimates at 60 Hz. We take the average of the estimated gaze of left and right eyes. Figure 1(b) shows the coordinate systems. The X axis corresponds to movement left and right. The Z axis corresponds to movement forward and backward. Barriers are placed to control the moving range of the subjects. According to the manufacturer's recommendations, the subject initially stands 60 cm away from the eye tracker along the blue dashed line in Figure 1(b) (52 cm along Z axis) and looks forward towards the center of a 17-inch monitor (33.72cm × 26.97cm with a resolution of 1280 × 1024). The height of the monitor can be adjusted to match the subject's height so that s/he will look at the center of screen when looking forward.

The first experiment investigated the influence of eye position and time on the eye tracking performance. Two subjects participated in this experiment, all with normal or corrected-to-normal vision. Each subject first calibrated the eye tracker at his/her initial position using the standard nine-point calibration provided by the manufacturer. The subject was then asked to stand at three different positions and to fixate at the same nine points used in calibration. The positions are 10cm apart along the X axis as shown in Figure2(a). Subjects repeated the nine-point fixation task at positions 1, 2, and 3 over three rounds (nine times in total).

Figure 2(b) and (c) show the results. Each point represents the mean gaze position when fixating on the stimuli positions in one task. Different colors represent different positions. We observe that the points corresponding to the same position but at different times are closely clustered, but that the clusters corresponding to different positions are more widely separated. This suggests that the eye gaze estimates degrade systematically with position, but are stable over time.

The second experiment investigated the eye tracking degradation with head movement for different subjects. Three participants (one male and two females, average age of 25.7, SD = 2.52) took part in this experiment. All subjects had normal or corrected-to-normal vision. All had experience in using eye trackers and had accuracy good enough for interactions of eye typing or computer control immediately after the manufacture's calibration at the suggested position.

At the start of each experiment, the subject calibrated the eye tracker using a standard nine-point calibration provided by the manufacturer at the initial position. The subject was then asked to perform two trials where they fixated at each of the points on the 5×5 grid shown in Figure 4 for 20 seconds. During one trial, the subject was asked to move left and right. During the other trial, the subject was asked to move forward and backwards. We collected around 30000 gaze data for each trial. We downsampled the gaze data to $12Hz$ and grouped them according to the subject's eye position in the horizontal/depth direction as reported by the Tobii eye tracker. $6\% \sim 30\%$ of gaze points were discarded because the eye tracker failed to measure the gaze, usually because the subject moved out of the eye tracker's trackbox or moved too quickly.

Figure 3 shows the error between the raw reported gaze and the stimuli position as a function of eye position. We observe that the dependency of the error on position for different subjects is quite different. For example, Subject 2 exhibits a much more rapid increase in gaze estimation error as s/he approaches the eye tracker than the other two subjects. This suggests that any corrections to the degradation should be done online for each subject independently. For positions outside the plotted range, we were unable to get reliable gaze estimates. Although the trackbox reported by the manufacturer is $44 \times 22 \times 30\text{cm}^3$(width $\times$ height $\times$ depth), we were able to get reliable tracking only over a smaller range.

3 TASK-EMBEDDED ONLINE CALIBRATION

Based on the conclusions from Section 2, we propose a position dependent linear homography-based method to correct the raw gaze estimates from the remote eye tracker. During the eye tracker operation, we collect a stored history of triples $(p_t, g_t, \hat{g}_t)$ at each gaze sample time t, where $p_t \in \mathbb{R}^3$is the measured eye position from the eye tracker in World Coordinate System, $g_t \in \mathbb{R}^3$ is the raw gaze estimate, and $\hat{g}_t \in \mathbb{R}^3$ is the gaze target position, both in homogeneous form in the Display Coordinate System.

For the current eye position p_t and measured gaze g_t, we apply a 3×3 linear transformation matrix $A(p_t)$ to correct the measured

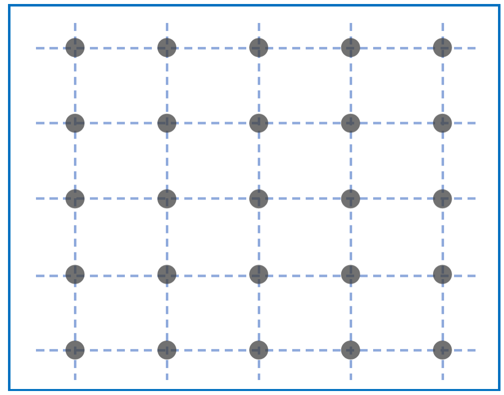


Figure 4: Positions of 5×5 grid over screen

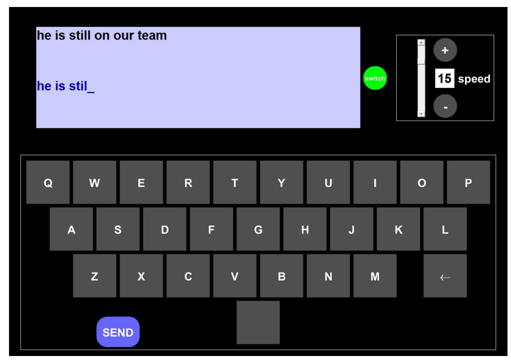


Figure 5: Keyboard layout.

gaze:

$$\tilde{g}_t = A(p_t) g_t. \tag{1}$$

$A(p_t)$ is estimated from the gaze points in the stored history by solving the following linear weighted regularized least square problem:

$$A(p_t) = \underset{A}{\operatorname{argmin}} \left(\sum_{i=1}^{t-1} w_i(p_t) \|\hat{g}_i - A g_i\|_2^2 + \lambda \|A - I\|_F^2 \right), \tag{2}$$

The regularized term is the Frobenius Norm, where

$$\|A - I\|_F^2 = \operatorname{Tr}\left((A - I)^T (A - I)\right). \tag{3}$$

This increases the robustness when we have very few reliable observations, by biasing the correction matrix towards the identity.

The weight $w_i(p_t)$ decreases with distance between the current eye position p_t and the eye position p_i of the ith triple in the stored history:

$$w_i(p_t) = e^{-\frac{\|p_t - p_i\|^2}{2\sigma^2}}, \tag{4}$$

This ensures that the history data from eye positions closest to the current position have the most influence.

The parameter λ determines the trust the algorithm has in the raw gaze estimates from the remote eye tracker. When $\lambda = \infty$, there is no online calibration ($A(p_t) = I$).

Equation 2 has a closed-form solution:

$$A(p_t) = (\hat{G} W G^T + \lambda I) * (G W G^T + \lambda I)^{-1}, \tag{5}$$

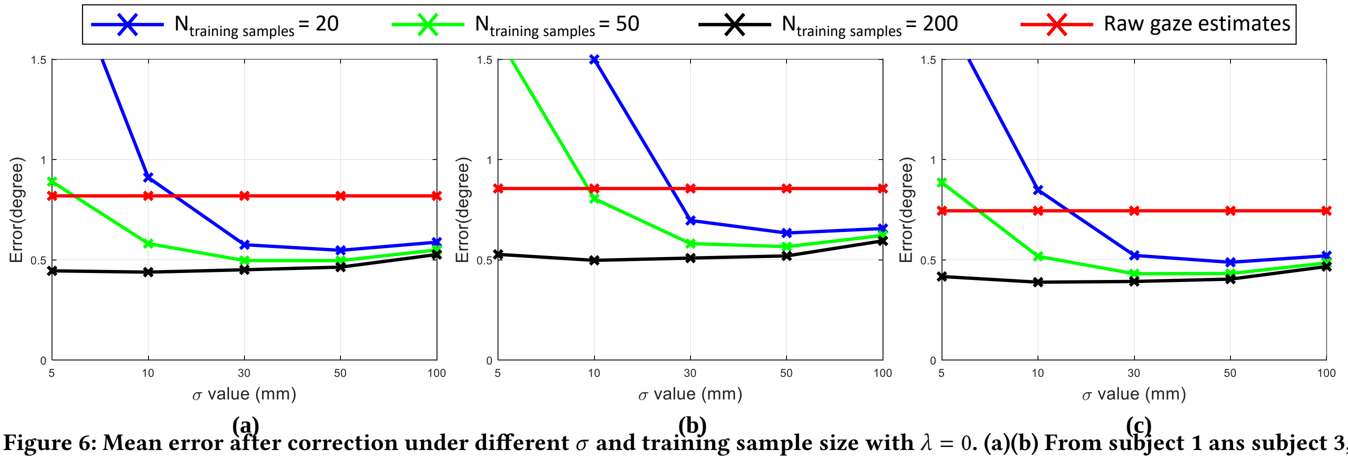


Figure 6: Mean error after correction under different σ and training sample size with $\lambda = 0$. (a)(b) From subject 1 ans subject 3, moving left and right. (c)From subject 2, moving forward and backward.

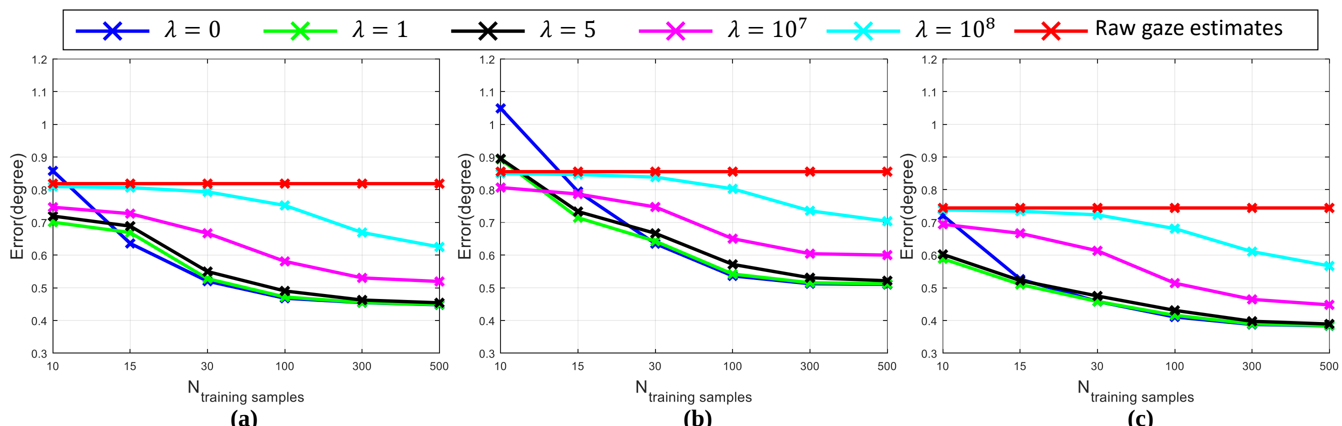


Figure 7: Mean error after correction under different λ and training sample size with $\sigma = 30mm$. (a)(b) From subject 1 and subject 3, moving left and right. (c)From subject 2, moving forward and backward.

where $G = [g_1, ..., g_{t-1}]$, $\hat{G} = [\hat{g}_1, ..., \hat{g}_{t-1}]$, and $W = diag(w_1(p_t), ..., w_{t-1}(p_t))$.

4 EXPERIMENTAL RESULTS

We describe the results of two experiments. The first experiment verifies that the proposed method can effectively correct the measured gaze at different head positions based on history observations. In the second experiment, we apply the proposed task-embedded online calibration method to dwell-based eye typing.

4.1 Within-subject gaze correction

In the first experiment, we evaluated the performance of our proposed method using the data collected in Section 2 for characterizing the subject-dependent sensitivity to head movement. We had two objectives: first to verify that the calibration algorithm can compensate for the degradation due to head movement and second to select values for σ and λ in Equations 2 and 4.

For each subject, we randomly chose 500 triples $(p_t, g_t, \hat{g}_t)$ as the testing set. We used the remainder of the data to generate the past history(training set) by uniformly sampling N triples across subject's moving space to calculate $A(p_t)$ according to Equation 5 in order to correct the raw gaze estimates in the testing set according to Equation 1.

To select the value of σ, we first set $\lambda = 0$, i.e., the calibration depends only on the gaze history. We then swept the value of σ from 5 to 100 for different gaze history sizes $N = \{20, 50, 500\}$. Figure 6 shows the results for the three subjects. We used the experiments of subject 1 and subject 3 when moving left and right and of subject 2 when moving forward and backward as typical examples. Similar curves are observed for the other cases. Since we uniformly sampled the training data across the subject's moving space, the larger the training size N, the more observations we have locally around for each gaze in the testing set, the smaller the optimal σ will be. We choose $\sigma = 30$mm since this led to the best performance considering different subjects and movement directions over a moderate history length ($N \approx 50$).

Given $\sigma = 30$mm, we evaluated the performance by sweeping the length of the stored history N over the range $\{10, 15, 30, 100, 300, 500\}$ for different values of λ, as shown in Figure 7. Similar curves were observed in all other experiments. Larger values of λ result in better performance for short history lengths N, by biasing $A(p_t)$ to the identity matrix. However, if λ is too large, the accuracy degrades

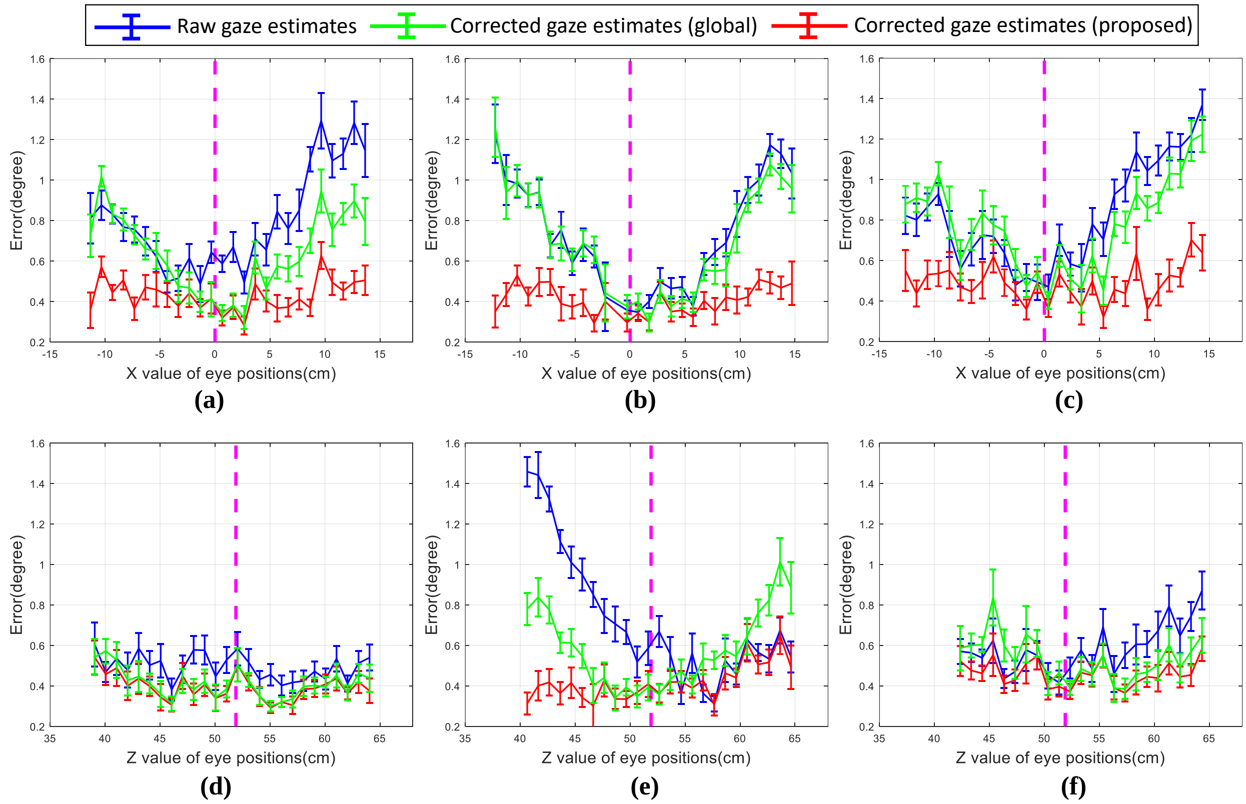


Figure 8: Results of correcting the raw gaze estimates for 3 different subjects. The first row shows the results from the three subjects when moving left and right. The second row shows the results from the three subjects when moving forward and backward. The blue line shows the accuracy of the raw gaze estimates. The red line shows the accuracy of the corrected gaze estimates. The green line shows the accuracy of gaze estimates corrected by a global (position independent) linear transformation. The vertical pink dashed line shows the initial position. Error bars indicate standard error. (a)(d) subject 1. (b)(e) subject 2. (c)(f) subject 3.

for larger values of N, since the information in the data is not fully exploited. For very large λ, there is no correction, since $A(p_t) = I$. We chose $\lambda = 1$, as it gave the best performance over most values of N for most subjects.

With $\sigma = 30$mm and $\lambda = 1$, we corrected all the raw gaze estimates in the testing set using a history length $N = 500$. Figure 8 compares the effect of calibration versus no calibration as a function of position, where we binned the gaze estimates according to eye position either horizontally or in depth. Uncalibrated (raw) gaze estimates vary widely with position, whereas the proposed method maintains good accuracy under large head movement. We also include the results of correcting the gaze by a global (position independent) correction matrix A computed by setting $w_i(p_t) = 1$, for all i and p_t. The proposed position dependent method achieves much lower error than using the global correction.

4.2 Online application for dwell-based eye typing

In this section, we illustrate the use of the proposed online calibration method in a dwell-based eye typing task. Figure 5 shows the interface of the keyboard layout. The horizontal distance between the neighboring keys and the vertical distance between rows are both 120 pixels. The space key is placed in the middle and below the letter keys. We also place a "SEND" key to indicate the completion of the current sentence entry. The stimulus sentence and transcript sentence are placed on the top panel. A switch button is placed at the right to enable/disable the keyboard. Online speed adjustment is also allowed through the speed adjustment panel. The typing is case insensitive. All keys have the same sized active selection area, a 120×120 pixel square centered at the key center. We have included visual and audio feedback in the experiment. After each selection, the key flashes in purple and a 'click' sound is generated. For each correctly typed character, we augment the gaze history by adding the triple $(p_i, g_i, \hat{g}_i)$, where p_i is the average measured eye position during the dwell time of a correct selection, g_i is the average raw gaze estimate, and $\hat{g}_i$ is the center position of the selected key. If the selection is wrong and is corrected later, the observation is removed from the training set. Unlike the evaluation in the previous section, which used a fixed history length randomly sampled across different head positions, this experiment is exactly consistent with the actual intended usage. The gaze history starts

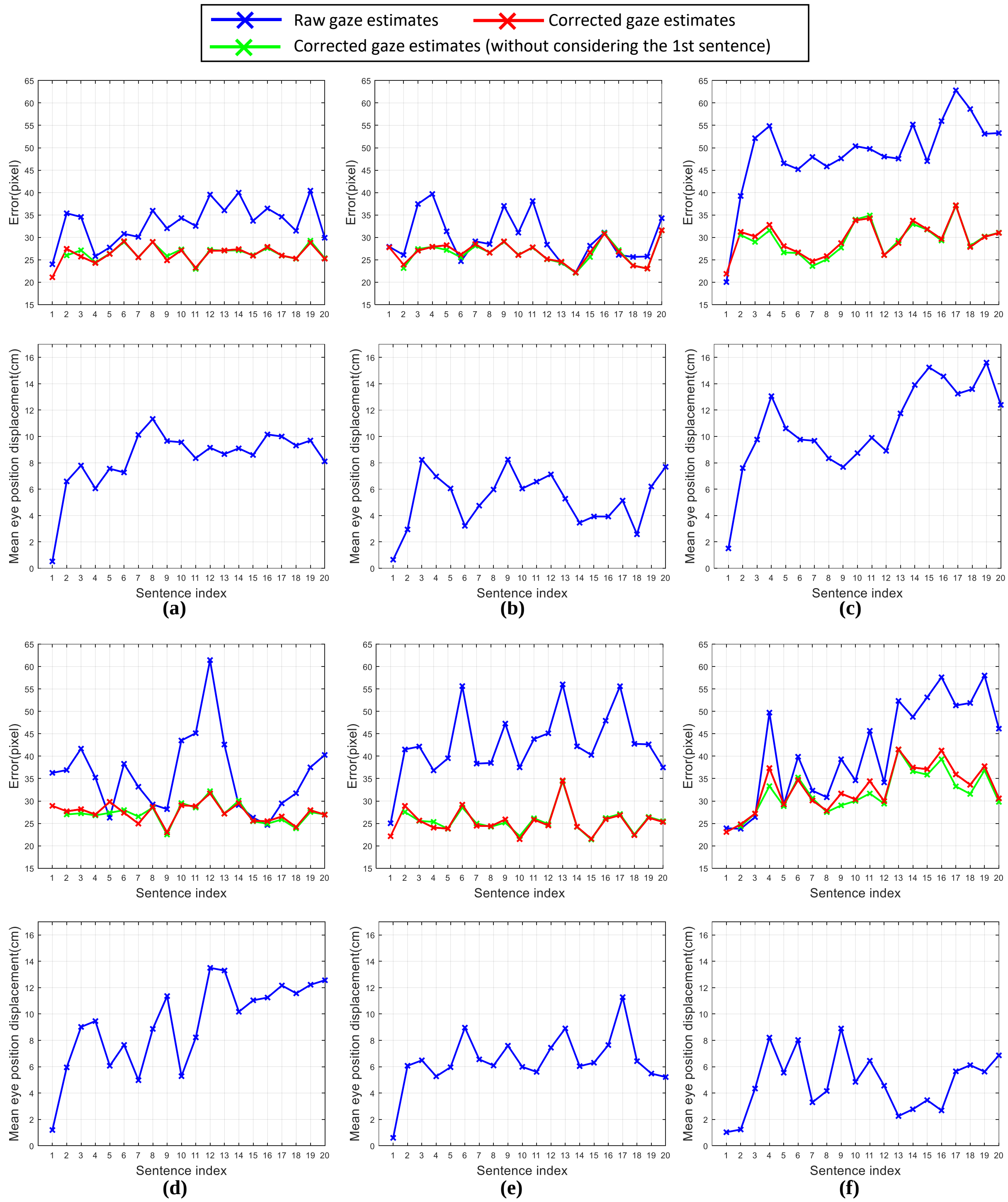


Figure 9: Result of online calibration in dwell-based eye typing. (a)-(f) show the result from the six subjects. The X-axis is the sentence index. The upper plots show the error as a function of sentence index. The red line shows the error of the corrected gaze estimates with our online calibration. The blue line shows the error of the raw gaze estimates. The bottom plots show the mean displacement between the eye position and the initial position while typing each sentence. For the angular error, because the angle is small, the error in degrees is approximately proportional to the error in pixels. In our setup, $\frac{Error(degree)}{Error(pixel)}$ is $0.018 \sim 0.019$. It varies with the eye position and gaze position on the screen.

out empty and grows over time. The gaze targets $\hat{g}_i$ are actually only estimates of the true gaze based on the task context.

Before each experiment, the subject calibrates the eye tracker using a standard nine-point calibration provided by the manufacturer at the suggested position (60cm from the eye tracker along the blue dash line in Figure 1(b)).

After initial calibration, the subject uses dwell-based eye typing to transcribe 20 sentences randomly selected from the phrase set described in [MacKenzie and Soukoreff 2003]. There is no requirement on typing speed, but subjects are asked to type as accurately as possible. Subjects are asked to stand still for typing the first sentence. After that, subjects are free to move naturally during the experiment, changing their head pose and standing pose and position. We only perform the experiments with the online calibration active, as during our initial exploratory experiments, we found that without online calibration the eye typing system would eventually fail due to miscalibration .

Limited by the CPU computational power, we can not solve Equation 2 online for too many gaze observations. To solve this problem, we use a fixed-size buffer to save the past gaze observations: 1) For each selection, we only save the mean gaze position within the dwell-time. 2) When the number of observations exceeds the buffer size, we simply discard earlier gaze observations. In this experiment, the buffer size is set to 1000. The time cost of our MATLAB code for correcting one gaze point based on 1000 gaze observations is about $0.008s$ using an Intel i5-4440 CPU. Since 1000 selections corresponds to about $10 \sim 29$ minutes of continuous text input with a speed of $6.90 \sim 19.89wpm$ [Majaranta et al. 2009], this should not limit the accuracy in practical situations.

Six participants (Three male and three females, average age of 26.2, SD = 2.48) took part in this experiment. All had normal or corrected-to-normal vision. All had experience in using eye trackers and have good accuracy of measured gaze immediately after the manufacturer's calibration at the suggested position.

Figure 9 illustrates the online calibration results of six different participants. We report the error in pixel here because the selection region of keys have fixed pixel size. The mapping from pixels to degrees changes with the eye position and gaze positions on the screen. Because the angle is small, the error in degrees is approximately proportional to the error in pixels. In our setup, if the subject stands centered directly in front of and 60cm away from the eye tracker the conversion factor from pixel to degree error is 0.019 when gazing at the center of the screen and 0.018 when gazing at the edge of the screen. For each correct selection, we average the Euclidean distance between gaze points within the dwell time and the center of the selected key as the error. We average the error over all correct selections in each sentence as the error of one sentence. For the first sentence, subjects are asked to stand very still. Thus, at beginning, the errors of both online calibration and measured gaze are small. Subjects are allowed to move naturally from the second sentence. Generally speaking the larger the displacement between the eye position and the initial position, the larger the error. However, the proposed online calibration method significantly reduces the error, resulting in high accuracy and fluid typing that remained stable over time. Considering that data collected during the first sentence may also act as additional initial calibration points since the subjects are standing still, we also simulated proposed online calibration with data starting from the second sentence. The results, shown as the green line in Figure 9, are very similar to those achieved when using all data.

5 DISCUSSION

Gaze estimation of remote eye tracking usually degrades as users naturally move during the usage. We propose an implicit online calibration as users are doing tasks. The online calibration is made by referencing to the past observations from which we have estimates of the gaze targets from the tasks. An eye-position dependent linear transformation matrix is derived to correct the measured gaze from the remote eye trackers. Experimental results on both a complete evaluation and a typical general usage of dwell-based eye typing show that the proposed method can significantly reduce the error when subjects introduce a large range of head movement.

Three assumptions are made in the proposed methods. First, we assume the estimated eye positions (not gaze estimates) from the eye tracker are accurate. Since we assign the weighting factors in computing the correction matrix $A(p_t)$ using the eye positions, the system will not work well if the eye position estimation is unreliable. Second, we assume the gaze target is at the center of the key when the subject is doing dwell-based selection. This is likely not true in practice, since the keys are relatively large, and the human fovea covers 1°visual angle. However, if these errors are small and unbiased, this may be compensated for by our least squares formulation, especially when the gaze history is long. Third, we also assume that natural head movements are slow movements. The more reliable observations we have, the better the correction will be. If the user slowly moves the head, the more he/she interacts, the better the online calibration will be as the user moves to larger and larger displacements from the initial position. However, if the user moves quickly to a position where there is no reliable history and the position is also far from the initial position, it is highly likely possible that the system will not correct the gaze estimate due to a lack of reliable history for calibration.

The proposed method is limited as it requires estimates of the gaze targets. However, we believe gaze target estimates are available for many screen-based interactions. For example, mouse clicks and keyboard interactions may also be used to estimate gaze targets, in addition to those obtained from the dwell-based selections.

Since our takes only raw gaze estimates without regard to how they were generated, we believe that it will be applicable to many existing eye trackers, both commercial and non commercial, as long as the assumption that the degradation in gaze estimation due to head position is systematic and stable over time. Our method assumes a simple position dependent linear transformation for correcting raw gaze estimates. Additional improvements may be possible by modifying the gaze estimation algorithms directly, however, the source code and algorithms used for calibration and gaze estimation are often not available. In the absence of more detailed information about how the gaze estimates are generated, our algorithm provides an effective and widely applicable way to reduce the degradation in eye tracking performance over time.

ACKNOWLEDGMENTS

This research was funded in part by the Hong Kong Research Grants Council under grant number 16211015.

REFERENCES

Jixu Chen and Qiang Ji. 2015. A probabilistic approach to online eye gaze tracking without explicit personal calibration. *IEEE Transactions on Image Processing* 24, 3 (2015), 1076–1086.

Zhaokang Chen and Bertram E Shi. 2018. Using Variable Dwell Time to Accelerate Gaze-Based Web Browsing with Two-Step Selection. *International Journal of Human–Computer Interaction* (2018), 1–16.

Peter M Corcoran, Florin Nanu, Stefan Petrescu, and Petronel Bigioi. 2012. Real-time eye gaze tracking for gaming design and consumer electronics systems. *IEEE Transactions on Consumer Electronics* 58, 2 (2012).

Flavio L Coutinho and Carlos H Morimoto. 2013. Improving head movement tolerance of cross-ratio based eye trackers. *International journal of computer vision* 101, 3 (2013), 459–481.

Xujiong Dong, Haofei Wang, Zhaokang Chen, and Bertram E Shi. 2015. Hybrid brain computer interface via Bayesian integration of EEG and eye gaze. In *2015 7th International IEEE/EMBS Conference on Neural Engineering (NER)*. IEEE, 150–153.

EyeFollower. 2019. EyeFollower @ONLINE. https://www.interactive-minds.com/eye-tracker/eyefollower

Dan Witzner Hansen, Javier San Agustin, and Arantxa Villanueva. 2010. Homography normalization for robust gaze estimation in uncalibrated setups. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. ACM, 13–20.

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.

Roy S. Hessels, Tim H. W. Cornelissen, Chantal Kemner, and Ignace T. C. Hooge. 2015. Qualitative tests of remote eyetracker recovery and performance during head rotation. *Behavior Research Methods* 47, 3 (01 Sep 2015), 848–859.

Anthony J Hornof and Tim Halverson. 2002. Cleaning up systematic error in eye-tracking data by using required fixation locations. *Behavior Research Methods, Instruments, & Computers* 34, 4 (2002), 592–604.

Michael Xuelin Huang, Tiffany CK Kwok, Grace Ngai, Stephen CF Chan, and Hong Va Leong. 2016. Building a personalized, auto-calibrating eye tracker from user interactions. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 5169–5179.

Qiang Ji and Xiaojie Yang. 2002. Real-time eye, gaze, and face pose tracking for monitoring driver vigilance. *Real-time imaging* 8, 5 (2002), 357–377.

Dongwook Jung, Jong Lee, Su Gwon, Weiyuan Pan, Hyeon Lee, Kang Park, and Hyun-Cheol Kim. 2016. Compensation method of natural head movement for gaze tracking system using an ultrasonic sensor for distance measurement. *Sensors* 16, 1 (2016), 110.

Anuradha Kar and Peter 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.

Hyeon Chang Lee, Duc Thien Luong, Chul Woo Cho, Eui Chul Lee, and Kang Ryoung Park. 2010. Gaze tracking system at a distance for controlling IPTV. *IEEE Transactions on Consumer Electronics* 56, 4 (2010).

I Scott MacKenzie and R William Soukoreff. 2003. Phrase sets for evaluating text entry techniques. In *CHI'03 extended abstracts on Human factors in computing systems*. ACM, 754–755.

Päivi Majaranta, Ulla-Kaija Ahola, and Oleg Špakov. 2009. Fast gaze typing with an adjustable dwell time. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 357–360.

Takashi Nagamatsu, Michiya Yamamoto, and Hiroshi Sato. 2010. MobiGaze: Development of a gaze interface for handheld mobile devices. In *CHI'10 Extended Abstracts on Human Factors in Computing Systems*. ACM, 3349–3354.

Diederick C Niehorster, Tim HW Cornelissen, Kenneth Holmqvist, Ignace TC Hooge, and Roy S Hessels. 2018. What to expect from your remote eye-tracker when participants are unrestrained. *Behavior research methods* 50, 1 (2018), 213–227.

Takehiko Ohno and Naoki Mukawa. 2004. A free-head, simple calibration, gaze tracking system that enables gaze-based interaction. In *Proceedings of the 2004 symposium on Eye tracking research & applications*. ACM, 115–122.

Jimin Pi and Bertram E Shi. 2017. Probabilistic adjustment of dwell time for eye typing. In *Human System Interactions (HSI), 2017 10th International Conference on*. IEEE, 251–257.

Wayne J Ryan, Andrew T Duchowski, and Stan T Birchfield. 2008. Limbus/pupil switching for wearable eye tracking under variable lighting conditions. In *Proceedings of the 2008 symposium on Eye tracking research & applications*. ACM, 61–64.

Yusuke Sugano, Yasuyuki Matsushita, Yoichi Sato, and Hideki Koike. 2015. Appearance-based gaze estimation with online calibration from mouse operations. *IEEE Transactions on Human-Machine Systems* 45, 6 (2015), 750–760.

Tobiipro. 2019. Does head movement affect eye tracking results? @ONLINE. https://www.tobiipro.com/learn-and-support/learn/eye-tracking-essentials/does-head-movements-affect-eye-tracking-results/

Haofei Wang, Jimin Pi, Tong Qin, Shaojie Shen, and Bertram E Shi. 2018. SLAM-based localization of 3D gaze using a mobile eye tracker. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*. ACM, 65.

Dong Hyun Yoo and Myung Jin Chung. 2005. A novel non-intrusive eye gaze estimation using cross-ratio under large head motion. *Computer Vision and Image Understanding* 98, 1 (2005), 25–51.

Yunfeng Zhang and Anthony J Hornof. 2011. Mode-of-disparities error correction of eye-tracking data. *Behavior research methods* 43, 3 (2011), 834–842.