CalibMe: Fast and Unsupervised Eye Tracker Calibration for Gaze-Based Pervasive Human-Computer Interaction

Thiago Santini
University of Tübingen
Tübingen, Germany
thiago.santini@uni-tuebingen.de

Wolfgang Fuhl
University of Tübingen
Tübingen, Germany
wolfgang.fuhl@uni-tuebingen.de

Enkelejda Kasneci
University of Tübingen
Tübingen, Germany
enkelejda.kasneci@uni-tuebingen.de

ABSTRACT

As devices around us become *smart*, our gaze is poised to become the next frontier of human-computer interaction (HCI). State-of-the-art mobile eye tracker systems typically rely on eye-model-based gaze estimation approaches, which do not require a calibration. However, such approaches require specialized hardware (e.g., multiple cameras and glint points), can be significantly affected by glasses, and, thus, are not fit for ubiquitous gaze-based HCI. In contrast, regression-based gaze estimations are straightforward approaches requiring solely one eye and one scene camera but necessitate a calibration. Therefore, a fast and accurate calibration is a key development to enable ubiquitous gaze-based HCI. In this paper, we introduce *CalibMe*, a novel method that exploits *collection markers* (automatically detected fiducial markers) to allow eye tracker users to gather a large array of calibration points, remove outliers, and automatically reserve evaluation points in a fast and unsupervised manner. The proposed approach is evaluated against a nine-point calibration method, which is typically used due to its relatively short calibration time and adequate accuracy. *CalibMe* reached a mean angular error of 0.59° ($\sigma = 0.23°$) in contrast to 0.82° ($\sigma = 0.15°$) for a nine-point calibration, attesting for the efficacy of the method. Moreover, users are able to calibrate the eye tracker anywhere and independently in ≈ 10 s using a cellphone to display the collection marker.

ACM Classification Keywords

H.1.2 User/Machine Systems: Human factors; H.5.2 Information Interfaces and Presentation: User Interfaces Auditory (non-speech) feedback, Ergonomics, Interaction styles, User-centered design

Author Keywords

Eye Tracking; Calibration; Gaze-Based Interaction; Fiducial Markers; Usability

CHI 2017, May 06-11, 2017, Denver, CO, USA

DOI: http://dx.doi.org/10.1145/3025453.3025950

INTRODUCTION

The human-gaze provides paramount cues for human communication and interaction [20]. Following this insight, gaze has been proposed as a HCI interface since at least the early 90s, with some believing that gaze will revolutionize the way we interact with our devices [26]. Gaze-based HCI in stationary scenarios (e.g., desktop computing) has matured enough that cheap and mainstream eye trackers (such as the Eye Tribe [39] for as low as $199 USD) have emerged. However, the gaze-based HCI in mobile and pervasive scenarios required for interaction with *smart*[1] devices remains an open challenge. For the remainder of this paper, we will focus on these mobile and pervasive scenarios. In particular, we concentrate on head-mounted video-based eye trackers as these are unintrusive, flexible, and mobile, making them excellent candidates for pervasive eye tracking [11]. These devices consist of a glasses-like frame with at least one camera capturing images of the user's eye and one capturing part of the user's field of view (see Figure 1a); gaze estimation is then the process of inferring the user gaze position in the field image based on the eye images.

(a) Pupil Labs eye tracker. (b) Google Glass.

Figure 1: A monocular eye tracker from Pupil Labs (a). It is worth noticing that some patents [33] hint at eye tracking capabilities being integrated in cutting-edge head-worn HCI devices such as the Google Glass (b).

A particular aspect of gaze estimation is the calibration step, which is used to produce a function mapping the position of the user's eyes to gaze. High-end state-of-the-art mobile eye tracker systems (e.g., SMI and Tobii glasses [15, 38]) rely on geometry-based gaze estimation approaches, which can provide gaze estimations without calibration. In practice, it is

[1]We use the term *smart* to refer to interactive network connected devices.

common to have at least an one point calibration to adapt the geometrical model to the user and estimate the angle between visual and optical axis. Additionally, it has been reported that additional points are generally required to achieve satisfactory accuracy [40]. Furthermore, such approaches require specialized hardware (e.g., multiple cameras and glint points), cost in the order of tens of thousands of $USD, and are susceptible to inaccuracies stemming from lens distortions [22]. On the other hand, mobile eye trackers that make use of regression-based gaze-mappings require a calibration step but automatically adapt to distortions and are comparatively low-cost (e.g., a research grade binocular eye tracker from Pupil Labs is available for $2340 EUR [23]). It is worth noticing that similar eye trackers have been demonstrated by mounting one eye and one field camera onto the frames of glasses [2, 25, 1], yielding even cheaper alternatives for the more tech-savy users.

In its current state, the calibration step presents some disadvantages and has been pointed out as one of the main factors hindering a wider adoption of eye tracking technologies [29]. Popular calibration procedures customarily require the assistance of an individual other than the eye tracker user in order to calibrate (and check the accuracy of) the system. The user and the aide must coordinate so that the aide selects calibration points accordingly to the user's gaze. As a result, current calibration procedures cannot be performed individually and require a considerable amount of time to collect even a small amount of calibration points, impeding their usage for ubiquitous eye tracking. Henceforth we will refer to these methods as *N-Points* calibrations, where *N* is the amount of calibration points employed.

In this paper, we propose a novel approach – dubbed *CalibMe* (*Calibrating with Movements*) – that enables users to quickly and independently calibrate the eye tracker based on the movement of *collection markers*. We define *collection markers* as automatically detected markers meant to dynamically collect large arrays of relationship points between a user's eye position and gaze for both calibration and evaluation. These markers can be contrasted with *calibration markers*, which are used as reference points in a more static fashion to collect a small amount of calibration points. We employ a specific ArUco [14] marker selected based on multiple properties that make it an efficient collection marker, compared to custom markers employed as calibration markers in previous work. This allows us to hijack an existing and well established fiducial marker detection method used for augmented reality and to define Areas of Interest (AOIs) to enable *CalibMe* without incurring additional and costly image processing. Employing a collection marker, users are able to collect a significant amount of eye-gaze relationships for calibration and evaluation in an unsupervised fashion by moving their heads or the marker while fixating the center of the marker. We then propose rationalized outliers removal approaches to automatically eliminate ill-conditioned samples as well as a parameterizable method for the automatic selection of evaluation points. Effectively, these operations enable the users to quickly calibrate and assess gaze estimation quality without the assistance of a supervisor[2], such as in the envisioned use cases illustrated in Figure 2. Moreover, the ramifications of allowing head rotation during calibration are discussed, and different collection movement patterns are proposed and evaluated. Additionally, the efficacy of *CalibMe* is compared to a typical *9-Points* calibration based on a regular twenty five point grid evaluation. *9-Points* was selected for evaluation as it presents a reasonable trade-off between accuracy and calibration time. *CalibMe* is integrated into *EyeRecToo* [34], an open-source data acquisition software for head-mounted eye trackers, and, thus, readily available. We also provide a companion *Android* application that the user can use to display the *collection marker* and collect eye-gaze relationships. *EyeRecToo* and the companion *app* can be downloaded at www.ti.uni-tuebingen.de/perception.

GAZE ESTIMATION AND RELATED WORK

As previously mentioned, *calibration-less* approaches exist. However, geometrical (or model-based) gaze estimation either requires specialized hardware or exhibits inferior accuracy w.r.t regression-based approaches. For instance, Swirski et al. [37] report a mean angular gaze error of $1.68° (\sigma = 0.34°)$[3] *for noiseless simulated data*, whereas regression-based models have been shown to produce mean angular gaze errors as low as $0.4°$ ($max. = 1.11°$) in realistic scenarios [7]. Moreover, these approaches produce inaccurate gaze estimations if the user wears corrective glasses [22] as distortions caused by lenses are not accounted for in the models. A second calibration-less alternative, appearance-based methods, rely on the pixel intensity of the eye images to produce a gaze estimation and is also known to produce large angular errors. For instance, Zhang et al. [42] report errors of $\approx 6.1°$ ($\sigma \approx 1°$) for a within-dataset leave-one-person evaluation of their convolutional neural network approach. For a detailed survey on gaze estimation methods, we refer the reader to the work by Hansen et al. [16]. For the remainder of this section, we will focus on regression-based methods, which require a calibration procedure.

Whereas there is a large amount of work investigating how to improve regression-based gaze estimation, previous work has mostly focused on investigating different regression approaches (e.g., polynomial fit [7], projective transformations [41], neural networks [9]) and optimizing their parameters (e.g., polynomial order, number of hidden neurons). Little attention has been given to improving the calibration procedure, which has remained largely unmodified since its inception. In general, reference points are placed as to cover the expected range of visual movements of the subject[4]. Afterwards, a supervisor and the subject cooperate to collect calibration points.[5]. The supervisor is also responsible for checking that eye features (mainly the pupil center) are detected correctly throughout the process as well as the gaze estimation accuracy after calibration.

[2] It is worth noticing that evidence suggests improved gaze estimation accuracy and precision when the participant has control over eye-gaze relationships collection [30].

[3] In this work, μ is the mean value and σ is the standard deviation.

[4] Natural features can also be used as reference points.

[5] The minimal amount of points is usually determined by mathematical constraints from the regression – e.g., the polynomial order.

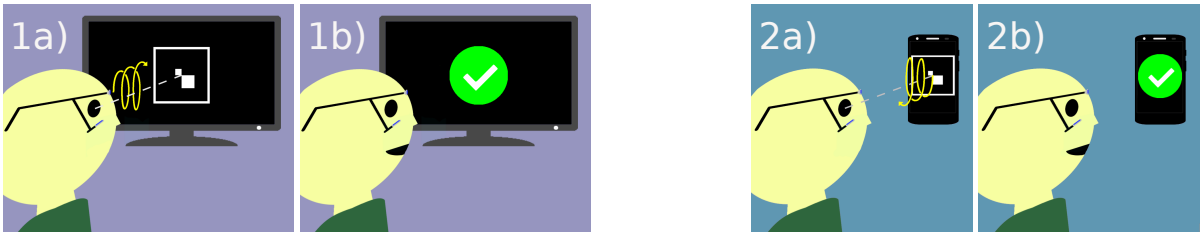


Figure 2: Illustrations of two envisioned *CalibMe* use cases. 1a) The user puts his head-worn eye tracker on, which detects the new user and requests the smart TV to display a collection marker; the user then fixates the marker and moves his head to collect eye-gaze relationships. 2a) The eye tracker detects its calibration became invalid (e.g., based on saliency-gaze overlap) and requests the smartphone to notify the user; the user then moves the smartphone displaying the collection marker to collect eye-gaze relationships. 1b and 2b) the eye tracker notifies the user that the calibration has been performed successfully through other smart devices or visual/haptic/audible feedback, signaling that the system is now ready to use for gaze-based interaction with other devices.

Atypical calibration approaches try to adapt the calibration procedure to their needs. For instance, Pfeuffer et al. [32] propose using objects moving with a known trajectory in a display to calibrate a high-range remote eye tracker placed under a screen, reaching mean accuracies of $\approx 0.6°$ ($\sigma \approx 0.1°$). In a similar stationary scenario, Huang et al. [19] proposed utilizing interactions events between the user and a computing system to collect calibration samples, reporting errors of 2.56°. In a driving scenario, Bernet et al. [4] employed a custom marker (consisting of two nested black squares) that is automatically detected, avoiding, thus, the need for the supervisor and user to coordinate. The user then fixates the custom marker and moves his head in steps of 10 cm exclusively in the horizontal and vertical directions since the proposed approach does not consider depth changes or head rotations. Unfortunately, reported results are based on simulations for their custom made eye tracker, and only the reprojection error for calibration points is given in degrees. In a distinct section (5.1), they report best results for evaluation points with a mean error of 2.22 px ($\sigma = 1.42$ px) at a distance of 2.5 m *for noiseless simulated data*; however, there is not enough information in the section to infer these values in terms of degrees. More similarly to the pervasive scenario, Evans et al. [10] compared two methods of collecting calibration points in outdoors environments: The first (*moving target*) consists of the user following a partner's thumb with his eyes; the partner then pauses the thumb at five distinct points, which are used for calibration. The second (*head tick*) consists of the users fixating a fixed point and moving their head in $\approx 10°$ steps in an asterisk-like pattern, producing about 25 to 30 calibration points. The collected points were later employed in an offline calibration, and both methods exhibited similar accuracies for central points with a mean error of 0.83° (σ not reported), but the *head tick* approach resulted in better estimations at points in the periphery. The authors also report that the *moving target* method was significantly faster than the head ticks. Pupil Labs employ cocentric circular markers in their *manual marker calibration* in a similar fashion to the *moving target* from [10]; similar restrictions also apply as mentioned in their website: "*this method is done with an operator and a subject*" [23].

It is worth mentioning that some previous works try to counteract calibration degradation through compensation or recalibration. Hornof et al. [18] employs implicit required fixation locations to evaluate the gaze estimation and correct systematic errors. Kolakowski et al. [21] attempt to isolate eye tracker drift based on the corneal reflection gain, which can then be filtered. Sugano and Bulling [36] use gaze input features and saliency maps calculated over the field camera images to (re)-calibrate the eye tracker. Lander et al. [24] perform a recalibration step with a subset of the initial calibration points; afterwards, the updated positions for calibration points not present in the recalibration are extrapolated. Binaee et al. [5] employ a set of ground-truth fiducial positions in a virtual environment to dynamically refine the calibration over time.

ON THE SELECTION OF COLLECTION MARKERS

A marker to be employed in the collection of eye-gaze relationships should have the following properties:

1. The user should be able to easily locate and distinguish the reference point to be fixated; for instance, points lying in the intersection of lines.

2. Marker detection should be accurate, precise, and require low resources since it must run in real-time in an embedded system alongside other eye tracking related image processing algorithms (e.g., pupil detection [12]).

3. Since the field camera moves w.r.t. the marker, small blurring effects are to be expected. The more robust to blur the marker, the faster the movements allowed, and, thus, the calibration process. However, one should be aware that the user gaze may lag behind the marker at higher velocities due to constrains in human smooth-pursuit capabilities, leading to less accurate gaze-marker relationships.

Hitherto, markers used for calibration follow a similar pattern: They tend to be custom bitonal, nested and cocentric shapes (see Figure 3a to Figure 3d). While such markers meet most of the requirements for a collection marker, they require a unique detection process. In contrast, fiducial markers designed for AOI definition and augmented-reality can meet these requirements and use a single process to detect a large set of markers.

Moreover, the detection of such markers is already integrated into most eye tracking software as AOI definition is a common and useful functionality in eye tracking experiments (e.g., ArUco markers are integrated into *EyeRecToo* and *Pupil Labs' Capture* [34, 23]). Therefore, we propose repurposing one of these markers to be used as collection marker; within the set of fiducial markers, it is likely that one exists, such that the stipulated collection marker requirements are met. In this work, we searched the default predefined ArUco dictionary used in *EyeRecToo* (*DICT_4X4_250*) for such a marker. Based on our requirements, we found marker #128 to be particularly suitable for the task. Furthermore, we increased the size of the black marker border from the default size to reduce blurring effects, resulting in the marker used throughout this work (Figure 3e). It is worth noticing that 249 markers remain for regular use in the set.

(a) (b) (c) (d) (e)

Figure 3: Markers used by Evans et al. [10] (a), Bernet et al. [4] (b), Pupil Lab [23] (c,d), and the ArUco marker (#128) selected for this work (e).

RATIONALIZED OUTLIERS REMOVAL

Consider a video-based head-mounted eye tracker with one eye camera and one field camera generating data at a predefined rate of r Hz. For each new eye tracker data incoming at timestamp (t), a pupil (p) is detected from the eye image, yielding the pupil center coordinates in the eye image (p_x, p_y) as well as its width (p_w) and height (p_h); similarly, the collection marker (cm) is detected from the field image, yielding its center coordinates in the field image (cm_x, cm_y). Let $D = \{t, p_x, p_y, p_w, p_h, cm_x, cm_y\}$ be the data tuple generated by the eye tracking system every $1/r$ seconds. The goal of the calibration procedure is then to collect data tuples containing pupil (p_x, p_y) and collection marker center (cm_x, cm_y) relationships in order to establish a function mapping pupil to gaze positions – i.e., the point of regard.

Intuitively, wrongly detected values for these variables will perturb the estimation of this function's parameters; thus, one of the supervisor tasks in the regular calibration is to check that the pupil is being detected correctly before association. For an extensive analysis of factors that may influence the correct pupil detection, we refer the user to the work by Fuhl et al. [13]. Additionaly, transient saccading and blinking during collection can be taken into account by collecting a position for a longer interval and taking the median of the samples. However, in an unsupervised calibration where the marker position can be constantly changing w.r.t. the user eye, these procedures are not possible. Thus, alternatives must be found. A common non-domain-specific approach is applying RANSAC to the fitting in order to eliminate outliers (e.g., as done by Bernet et al. [4]). Nonetheless, only outliers that significantly affect the fit are indentified, and, if data is particularly noisy, RANSAC will nonetheless return a fit. Additionally, randomly selecting subsets of points and selecting the best fit may not result in a good transformation function at all. Instead, we propose a series of rationalized approaches to remove outliers based on domain specific assumptions regarding head-mounted eye tracking setups, data, and algorithms; these outlier removals are described in the sequence, and examples are given in Figure 4.

Subsequent Pupil Size Ratio

During calibration, pupil size may change due to physiological factors, or the apparent pupil size in the image may change due to the eye position w.r.t. the camera. Nonetheless, the pupil size for two subsequent data tuples can be expected to remain largely the same due to the difference in magnitudes between the camera frame rate and pupil constriction/dilation speed. Additionally, the apparent size should also remain largely unchanged as the eye pursues the collection marker since no significant eye movement is elucidated. Thus, significant changes in pupil size can be attributed to false pupil detections; an example of outliers detected by this approach are sporadic detections of the iris as pupil – note that the center of the pupil and iris are not necessarilly at the same location [31].

Converging Pupil Position Range

Eye cameras are usually placed such that the whole eye, including canthi, is visible. Nonetheless, the pupil position is only expected to fall within a certain range within this image. Certain outliers will evoke pupil detections outside of this range; for instance, when the user blinks, the pupil detection algorithm may sporadically detect glasses frames, make-up, or moles as valid pupils. This outlier removal works by assuming the pupil positions to be normally distributed. Initially, all samples are considered inliers; then, this method computes the mean (μ) and standard deviation (σ) of all inliers, marking as outliers samples falling outside of the range $\mu \pm 2.7\sigma$ (i.e., covering $\approx 99.3\%$ of the distribution). This process is repeated until the amount of inliers converges.

Pupil Detection Algorithm Awareness

Some pupil detection algorithms consist of a main and a fallback method. Such fallback methods are typically employed when the pupil is in unfavorable positions for detection and tend to improve the recall of the algorithm (in terms of pupil detections) at the expense of a loss in accuracy of the pupil coordinate, shape, or orientation. For instance, the fallback mechanism of ElSe [12] (which is employed in this work) consists of searching for a point within a dark region and a strong center surround response without providing information about pupil shape and orientation. Thus, this outlier removal depends on the pupil detection method used and consists of requiring that the tuple has a valid pupil size to employ samples solely from the main method.

AUTOMATIC SELECTION OF EVALUATION POINTS

One of the main advantages of *CalibMe* is regarding the high amount of eye-gaze relationships that can be collected in a short period of time. After outliers removal, it is expected that

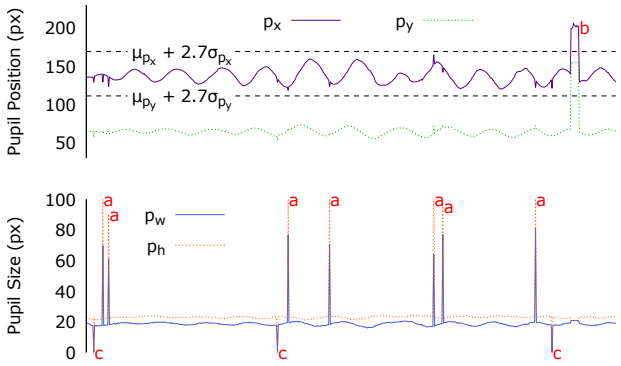


Figure 4: Rationalized outlier removal examples during a calibration of ≈ 21 s. Subsequent pupil size ratio outliers are identified by the letter a, converging pupil position range outliers by b, and pupil detection algorithm awareness by c. Notice how the pupil position estimate (p_x, p_y) is significantly corrupted by such outliers.

only valid data tuples are left. While the majority of these samples should go towards calibration in order to improve the regression, enough data tuples are produced that we can afford to exclude some samples from calibration to evaluate the resulting gaze estimation afterwards. In this section, we define a parameterizable method to select these evaluation points automatically.

The proposed selection method is defined by four parameters: the granularity (g), horizontal stride (Δ_x), vertical stride (Δ_y), and evaluation point range factor (rf). According to the first three parameters, a lattice is built around the center (f_x, f_y) of the field image; the lattice is defined by the points $(f_x - g \times \Delta_x, f_y - g \times \Delta_y)$ and $(f_x + g \times \Delta_x, f_y + g \times \Delta_y)$, with points laid down at every (Δ_x, Δ_y) stride inside this region. Afterwards, an elliptical area with a horizontal radius of Δ_x / rf and vertical radius of Δ_y / rf is associated with each lattice point, resulting in a configuration similar to that of Figure 5. After outliers removal, for each lattice point the remaining collected tuples are searched to find the tuple lying inside the associated elliptical region with minimal collection marker center distance from the lattice point. Each tuple found this way is then selected for evaluation. Afterwards, all tuples with a collection marker center matching those selected for evaluation are removed; in this manner, no evaluation point is used in the calibration regression. The resulting evaluation based on these points yields two metrics: 1) the reprojection error for evaluation points, which measure the *calibration accuracy*, and 2) the ratio of lattice points that have an evaluation point assigned to it, which represents the *calibration coverage*.

It is worth noticing that the evaluation points are likely to be spatially close to calibration points. The closer these points are, the more biased to superior results the evaluation is likely to be since the residuals for calibration points are minimized. Nonetheless, contrary to typical *N-Points* calibrations, *CalibMe* collectes a large and sparse amount of points for regression; combined with the low order polynomials commonly used in eye tracking, this results in the minimization being spread over the interpolation area instead of concentrated on a few points. Furthermore, even if evaluation points were to be selected independently from the calibration, these points are likely to fall nearby calibration points if the employed calibration pattern covers an adequate range of the field image. Moreover, as shown in the following section, collecting evaluation points independently from the calibration process not only requires a second collection process (and, thus, additional time), but these points are unlikely to fall on the calibration surface, therefore biasing the evaluation results to inferior results due to parallax error effects.

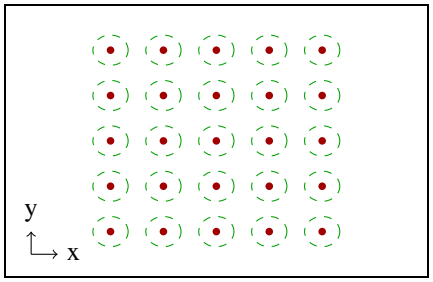


Figure 5: Evaluation lattice example ($g = 2$, $\Delta_x = 7°$, $\Delta_y = 6°$, and $rf = 3$) on a field of view of $56° \times 42°$.

ON CALIBRATION MOVEMENT PATTERNS

Prior to the analysis of different calibration movement patterns, it is paramount to elucidate 1) how the proposed calibration approach differs from a typical calibration, 2) what is the effect of allowing free head movements, and 3) how this affects the resulting evaluation. To illustrate these concepts, consider Figure 6, which shows the side view of a head-mounted eye tracking setup. Notice that the user's eye and the head-mounted field camera lie at different heights, which tends to be the common case[6]. Typically, the system is calibrated by employing calibration points in a planar surface (i.e., the *calibration plane*); within this plane (e.g., the white dot), the gaze estimation is the most accurate. Whereas on an ideal stationary setup (e.g., a computer screen) this can be expected, it is an unrealistic expectation for pervasive and mobile scenarios since the interactive objects will rarely lie on the calibration plane (e.g., the black dot). When the *object plane* differs from the *calibration plane*, the gaze estimation will produce an inaccuracy proportional to the distance between the calibration plane and object plane because the eye and field camera view the scene from a different angle, resulting in the *parallax* error [10, 17]. Moreover, notice that if the user is only allowed to move his head vertically and horizontally while fixating a stationary target, the resulting calibration is equivalent to that of a plane. On the contrary, if the user is allowed to change depths or rotate his head, the result can be seen as a *calibration surface* (illustrated as a rotation around the center of the camera in Figure 6). From this, three conclusions follow:

[6]For simplicity, we will not include the discrepancies when the camera is unaligned horizontally w.r.t the eye, but analogous effects result.

1. The underlying regression model should take into account that the calibration surface is not planar. Effectively, this is already the case since most current models employ curved relationships in order to compensate for lens distortions and eye curvature.
2. Relative to the gaze estimation based on a planar calibration, the gaze estimation based on a surface calibration may exhibit a larger or smaller parallax error depending on the object position and surface curvature.
3. Attempting to evaluate a surface calibration on a plane (or vice versa) *will bias the resulting accuracy due to the introduced parallax errors*. In fact, this applies for any two distinct calibration and evaluation surfaces (e.g., two distinct planes).

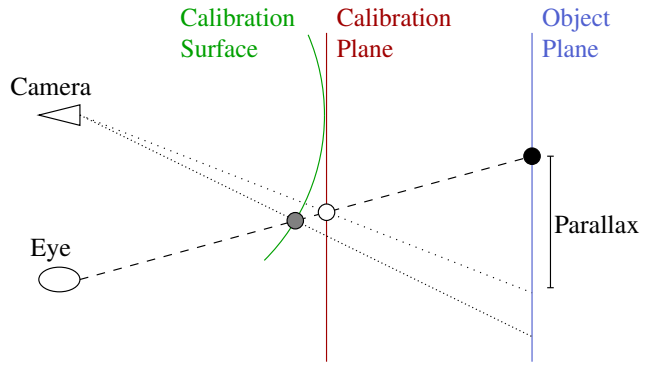


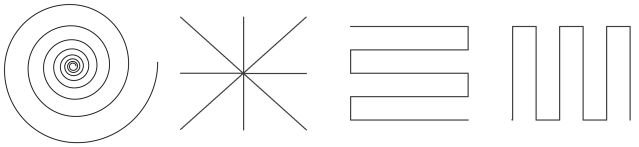
Figure 6: Parallax effect illustration between a curved calibration surface, a straight calibration surface (i.e., a plane), and the object plane.

As previously mentioned, one of the advantages of collecting evaluation points simultaneously with the calibration is the fact that the calibration and evaluation points will lie roughly on the same surface. If evaluation points were to be collected in a separate step, it is unlikely that the user will be able to reproduce the exact head and eye displacement as the one performed during calibration. Therefore, the gaze estimation is deteriorated due to the resulting parallax errors, producing an underestimation of the calibration quality. Hence, we evaluate the investigated collection movement patterns using the method proposed in the previous section.

Initially, we considered several continuous collection movement patterns for evaluation, such as a spiral, star, horizontal path, vertical path (shown in Figures 7a-d, respectively) as well as letting the user move freely. However, it quickly became apparent during a pilot study[7] that there are some properties that constitute superior patterns, recalling that the user must "draw" these patterns with the marker using the view of the field camera as "canvas":

1. The pattern should have intuitive parameters: While the pattern itself is usually clear from the illustration, most subjects were confused by parameters – e.g., how many horizontal lines in pattern Figure 7c should be performed. Curiously, no questions were asked regarding the parameters of the spiral pattern (Figure 7a).
2. The initial position of the pattern should preferably be in the center to allow starting in a natural position.
3. The extremeties of the field should be covered without necessitating precise movements from the user.

These criteria eliminated all but the spiral and star patterns; the spiral pattern is original from this work, whereas the star pattern is similar (albeit without pauses) to the *head ticks* pattern employed by Evans et al. [10]. The final form of the new pattern that we proposed for novices is a spiral movement, starting at the center, going outwards, and then spiraling back to the center. This is the spiral used in this work unless explicitly mentioned otherwise.



(a) Spiral (b) Star (c) Hori. Path (d) Vert. Path

Figure 7: Examples of movement patterns that can be employed in combination with collection markers.

EXPERIMENTAL EVALUATION

In this section, first the two investigated collection movement patterns are evaluated against each other using the proposed automatic evaluation point selection. Afterwards, the best performing one is compared against a typical *9-Points* calibration based on the proposed automatic evaluation point selection and a regular twenty five points grid evaluation.

Participants, Apparatus, and Metrics

Experiments were managed by an expert with more than one year of experience in conducting mobile and stationary eye tracking experiments. Five adult subjects participants took part in the evaluation (4 male, 1 female), and two of them wore glasses during the experiments. The subjects were briefed about the procedure, including in regard to head velocity and the goal of collecting the marker in multiple locations w.r.t. the field camera. The subjects were asked to verbally communicate when they were finished performing the movement, and *no instructions were given in regard to the collection duration for each pattern* as to not introduce artificial limits to the collection timing.

The experiment was conducted using a Dikablis Pro eye tracker [8]. This device has two eye (@60 Hz) and one field (@30 Hz) cameras; data tuples were sampled based on the frame rate of the field camera. The field camera was equipped with a 1.5× wide turn lens; camera parameters were estimated for use in the marker pose estimation, but the field image was not undistorted. *CalibMe* was integrated into *EyeRecToo* [34],

[7] During the pilot, the aim of the experiment was explained to five subjects, after which they performed the free-form movement; the users were then shown illustrations of each pattern in a random order and asked to perform those movements.

which was used to record and conduct the experiments. Pupil detection was performed using ElSe [12], and a bivariate second order polynomial regression was employed for gaze estimation in all cases. The poster shown in Figure 8 was used as stimuli and placed at a distance of approximately 1.1 m from the participants. This poster was designed for the reference points to cover about $40° \times 30°$ considering that after this range head movements become a regular feature of gaze shifts [3]. The eight red points in the extremities and the center of marker #128 were used as calibration points for the *9-Points* method. Blue dots and the center of marker #128 were used for the twenty five points evaluation. *CalibMe* automatic evaluation point selection configuration was set to match the setup of these evaluation dots ($g = 2$, $\Delta_x = 7°$, $\Delta_y = 6°$, and $rf = 3$). The metrics employed in this evaluation are:

Mean Angular Error (ε): evaluated as the mean of the euclidean distances between the evaluation point coordinates in the field image and the resulting coordinates from the gaze estimation.

Calibration Time (τ): evaluated as the timestamp difference between the last and first collected calibration tuples.

Pattern Coverage (γ): evaluated as the ratio between lattice points with an associated evaluation point and the total count of lattice points (25 in this study); this metric is only meaningful for *CalibMe*.

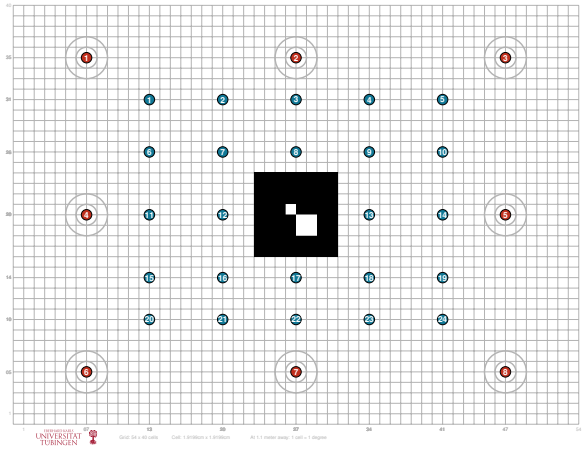


Figure 8: The calibration poster used during experiments and placed at 1.1 m away from the subjects. The red points cover an area of $\approx 40° \times 30°$ and are used for the *9-Points* calibration together with the center of marker #128. Blue points are employed for evaluation together with the center of marker #128 and lie within the interpolation area of the *9-Points* calibration.

Collection Movement Pattern Comparison

Each participant repeated each pattern three times, and no significant differences between repetitions were found; a visualization of resulting collection marker coordinates on the field image for one of the participants is shown in Figure 9. For the inter pattern comparison, we have aggregated all repetitions. No significant differences were found between the *Spiral* and the *Star* patterns in terms of calibration time ($F(1,28) = 0.690$, $p = 0.413$). However, the *Spiral* produced significantly larger coverage ($F(1,28) = 34.908$, $p = 0.0000023$). While no significant differences in terms of angular error were found ($F(1,28) = 3.486$, $p = 0.0724$), Figure 10 suggests the *Spiral* to have a small advantage over the *Star* in this regard. Additionally, this figure also exhibits the outcome in case no outliers removal is performed. Particularly interesting in this case is the single angular error outlier above 3.5°. This large error stems not from a bad estimation but from an outlier selected as evaluation point, demonstrating the importance of performing the outlier removal *before* points are automatically selected for evaluation.

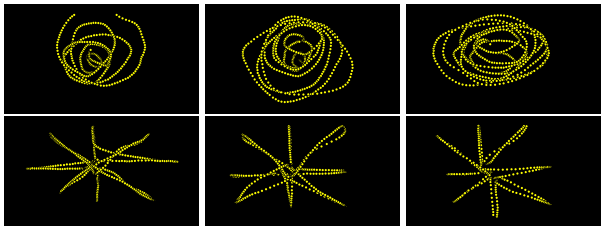


Figure 9: Collection marker center coordinates on the field image for one of the subjects when performing the *Spiral* (top) and *Star* (bottom) patterns repetitions.

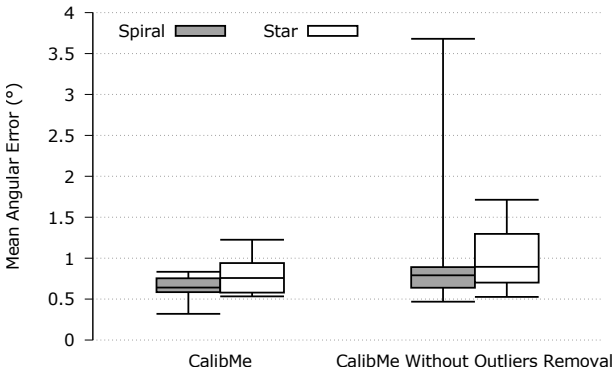


Figure 10: Accuracy for the *Spiral* and *Star* movement patterns measured through *CalibMe*'s automatic evaluation point selection with and without outliers removal.

CalibMe and 9-Points Calibration Juxtaposition

After performing the previously described experiment, each subject participated in a second phase, whose aim is to juxtapose calibrations performed with *CalibMe* against a typical *9-Points* calibration in a setup as similar as possible. Initially, the user rested his head on a chin rest placed at 1.1 m from the calibration poster shown in Figure 8. During this experiment, subjects were instructed not to communicate verbally in order to minimize head movements. The user then performed a *9-Points* calibration, followed by a twenty five points evaluation. The experimenter was responsible for manually selecting

the gaze points in the field camera view. After a point was selected, data tuples were collected for 500 ms, and the median of the collected pupil coordinates was associated with the gaze position. Following this step, the system produced an audible feedback so the user knew when to move to the next point, minimizing the amount of communication between experimenter and subject, thus making this calibration procedure faster and less error prone. After both procedures were finished, the experimenter slowly moved the subject's head away from the chin rest, moved the chin rest horizontally out of the way of the subject, and then moved the subject's head to the initial position by aligning it at a distance with the chin rest. With the head now free to move, the subject was instructed to perform the *Spiral* collection movement pattern.

As shown in Figure 11, this procedure results in two distinct surfaces – 1) *Pattern:* a surface formed by the sparse points collected while performing the *Spiral* pattern, and 2) *Poster:* a planar surface along the calibration poster (Figure 8). Therefore, a direct comparison between the regular and the collection method calibrations is *biased*. If we evaluate these solely on one of the surfaces, the accuracy of the other will be underestimated due to parallax errors (and vice-versa). Hence, we juxtapose these by cross-evaluating sets of calibration points from each surface on sets of evaluation points lying on both surfaces instead.

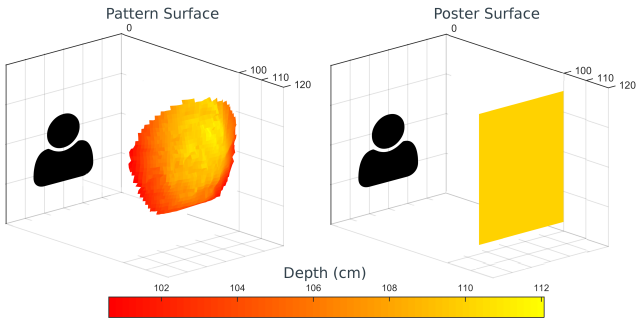


Figure 11: Surfaces produced by calibrating using a spiral head movement pattern and a regular *9-Points* calibration.

The following sets of calibration points were analyzed in this manner:

- *CalibMe*: this set consists of samples collected during the spiral movement, excluding the tuples selected for evaluation, tuples with collection markers at the same position as evaluation ones, and outliers identified by *CalibMe*. These samples lie on the *pattern surface*.
- *Outliers*: *CalibMe* without outliers removal.
- *9 Points:* the eight (red) calibration points plus the marker center; these samples lie on the *poster surface*.

These calibration sets were evaluated on two different evaluation sets:

- *Pattern (Reserved):* samples lying on the *pattern surface* that were automatically reserved for evaluation by *CalibMe*.
- *Poster (25 Points):* the twenty four (blue) evaluation points that lie on the *poster surface* plus the marker center.

The resulting gaze estimation accuracies from these evaluations are shown in Figure 12. As expected, both methods exhibit better accuracy when evaluated on their respective calibration surface than in a different one. The gaze estimation error due to the parallax effect can be noticed when comparing the same calibration set across different evaluation surfaces, which shows $\approx 0.7°$ of error in all cases. While this is in line with the expected error magnitude given our setup, it is worth noticing that part of these errors are contributed by occasional points that fall outside of the calibration interpolation range, for which the gaze estimation is expected to have an inferior accuracy. When comparing *CalibMe* against the *9-Points* calibration evaluated on their respective surfaces, no significant difference was found in terms of accuracy ($F(1,8) = 3.372$, $p = 0.104$). Nonetheless, this figure suggests that *CalibMe* produces slightly better results, yielding a mean angular error averaged over all participants of $0.59°$ ($\sigma = 0.23°$) in contrast to $0.82°$ ($\sigma = 0.15°$) for the *9-Points* calibration, attesting for the efficacy of the proposed approach.

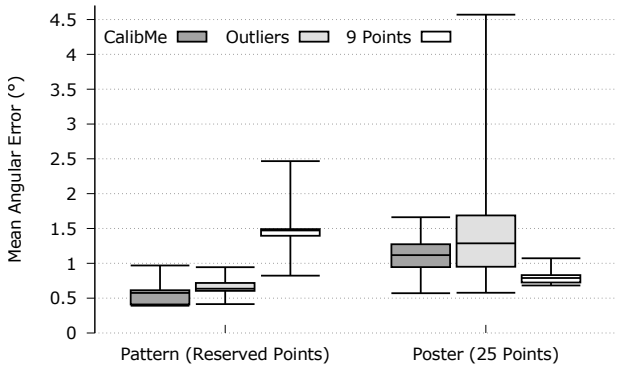


Figure 12: Mean angular error evaluated on points lying on the pattern and poster surfaces.

While no significant differences were found when comparing *CalibMe* and its outliers inclusive counterpart (*Outliers*), when visually inspecting the figure an anomaly (with a mean error larger than 4.5°) is clearly visible in the *25 Points* evaluation. By virtue of *CalibMe*'s automatic evaluation point selection, we quickly traced this anomaly as a result of a low coverage and a single pupil detection outlier in the right eye during a blink. This particular subject had a coverage ratio of 68%, whereas all other subjects had coverages of at least 80% ($\mu = 87\%$). As shown in Figure 13, this subject did not cover the left part of the 25 points evaluation area, resulting not only in a subpar accuracy, but also in no polynomial regression constraints in that area. The aforementioned outlier happens to lie in this unconstrained area, thus significantly skewing the gaze estimation at those points and artificially amplifying the mean angular error.

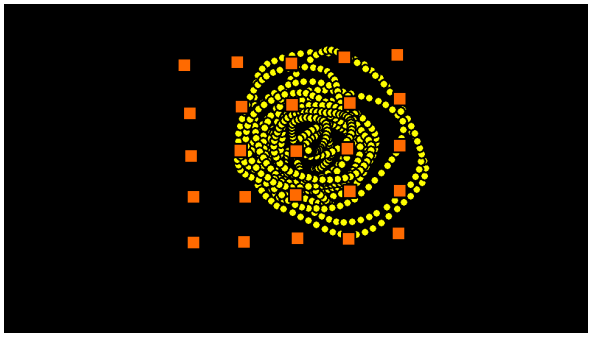


Figure 13: Collected tuples (yellow circles) and the 25 evaluation points (orange squares) for the subject with anomalous mean angular error when outliers are considered. Notice how the left side of the evaluation points are not covered by collected tuples.

A Note on Calibration Time

Assuming a qualified and experienced supervisor, the *9-Points* calibration time is rather constant if nothing disrupts the collection (e.g., in case one position must be repeated, the calibration flow is broken, and calibration time increases significantly). We can model the optimal time required for such calibrations by summing up 1) the sampling time for each point, 2) the time for the subject to react to the audible feedback and saccade to the next target, and 3) the time for the supervisor to react to the saccade and start the next sampling. Empirically, we found that the time to collect a single points in our setup is ≈ 1.67 s – i.e., one can expect about 15 s and 42 s to collect nine and twenty five points, respectively.

As previously mentioned, we intentionally did not impose time limitations on the subjects when performing the pattern calibration as to not bias results. Intuitively, the pattern calibration time depends on the spatial distribution and amount of collected tuples. These can be mainly determined by two factors: First, the user's skills and internal mental model of the system; for instance, the better a user can abstract the marker position w.r.t the camera, the more efficiently he can utilize his collection time. Second, the speed of the marker relative to the camera/subject; the faster the marker moves, the faster the calibration. However, several elements influence the latter factor, such as the camera resolution, frame rate, shutter type, marker detection blur robustness, and human smooth-pursuits physiological limitations. These elements are further discussed in section *LIMITATIONS*.

Since estimating user's skills is rather subjective, and finding users of different skill levels is unpractical, we approach this analysis from an alternative perspective. By downsampling the data from the previous experiment, we can study the effect of the amount of tuples on the method's accuracy. Here, downsampling is performed by using only one tuple in each *down sampling factor* (*DF*) tuples so that spatial distribution is preserved. It is worth noticing that a reduced amount of samples can compromise the coverage ratio and, thus, limit the information provided by the automatic evaluation. Therefore, we perform evaluations against the twenty five points instead. Figure 14 shows the effect of this downsampling in the mean angular error. It is clear that subjects collected an amount of tuples ($\mu = 549.8, \sigma = 159.11$) much larger than required to reach satisfactory accuracy. Thus, the limiting factor becomes the spatial distribution of the collected tuples and, consequently, the speed of the marker relative to the camera/subject. Assuming a conservative *DF* of two, the proposed approach already becomes significantly faster than the *9-Points* calibration, requiring on average 10.12 s as shown in Figure 15.

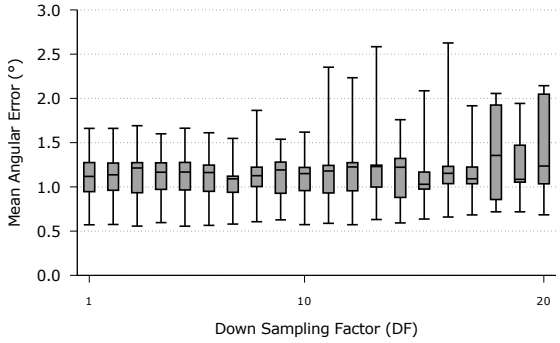


Figure 14: Downsampled *CalibMe* evaluated on the twenty five point grid, showing that downsampling by small factors retains accuracy as long as the spatial distribution is preserved.

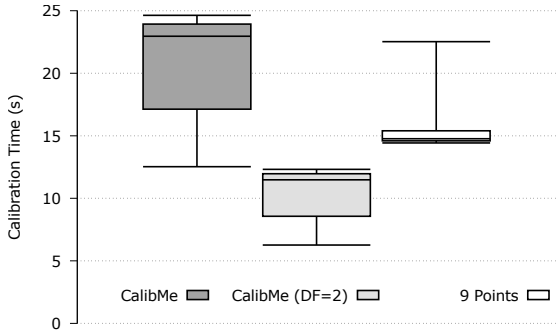


Figure 15: Calibration time for *CalibMe*, *CalibMe* downsampled by a factor of two, and the *9-Points* calibration.

In order to corroborate this time estimate, we conducted a separate experiment with a user considered experienced enough to have a good mental model of the marker movement relative to the camera: one of the *CalibMe* developers. As previously mentioned, a small amount of collected tuples may compromise the coverage ratio, limiting the amount of information provided by *CalibMe*'s automatic evaluation. Therefore, the user first collected eye-gaze relationships to be used for evaluation by staring into the center of a collection marker displayed in a cellphone screen and moving the cellphone in a grid pattern, as shown in Figure 16a. This collection took ≈ 43 s,

and tuples identified as outliers were removed from the evaluation set, resulting in 816 evaluation tuples. Afterwards, the user conducted ten independent *CalibMe* calibrations by moving the cellphone in spiral patterns, which were evaluated against the aforementioned evaluation tuples. It is worth noting that the user opted to perform a single outward spiral instead of the outward-inward pattern recommended for novices; instances of the resulting patterns can be seem in Figures 16b-16d. The average calibration time for these calibrations was 10.68 s ($\sigma = 0.86$ s), reaching an average angular error of 0.69° ($\sigma = 0.044°$). The individual values for each collection are shown in Figure 17. The result from this experiment is in line with both the expected angular error and calibration time, thus endorsing the time estimate of the downsampling analysis.

(a) (b) (c) (d)

Figure 16: The points used for evaluation (16a) and points from three (out of ten) distinct calibrations performed by the user (16b-16d).

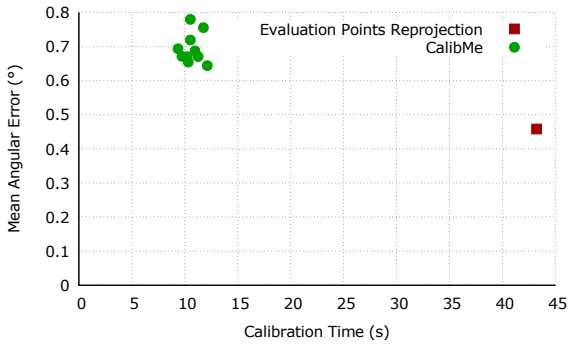


Figure 17: Calibration time and resulting mean angular error for the ten calibrations performed by the user; also shown, is the calibration time and mean angular error for the reprojection of the evaluation points.

LIMITATIONS

Naturally, the key limitation for *CalibMe* is in regard of marker and pupil detection. These tasks face similar challenges such as occlusion, highly skewed viewing angles, poor or irregular illumination, and motion blur [12, 6]. Nonetheless, there is significant research for these tasks in pervasive and challenging scenarios (see [13, 6]), and spurious incorrect detections during calibration can be eliminated through the proposed rationalized outliers removal methods. In particular, motion blur could be greatly alleviated by the usage of global-shutter sensors.

Aside from these detection challenges, human smooth-pursuit physiology imposes a lower bound on the calibration time. As marker speed increases, the smooth-pursuit gain (i.e., the ratio between the marker and smooth-pursuit speed) starts deviating significantly from one [27]. As a result, the marker center may not correspond exactly to the true gaze position, specially at higher speeds. These periods could potentially be identified based on real-time eye movement detection algorithms such as *I-BDT* [35].

FINAL REMARKS

In this work, we have introduced *CalibMe*, consisting of a collection marker that does not incur any additional processing to common eye tracking systems and a set of techniques to 1) quickly collect eye-gaze relationships for calibration, 2) remove ill-constrained relationship outliers, and 3) automatically reserve tuples for evaluation. As a result, *CalibMe* allows eye tracker users to quickly calibrate the system anywhere without supervision, giving feedback not only in terms of gaze estimation accuracy, but also on the calibration area coverage relative to the field camera. The proposed method reached accuracies ($\mu = 0.59°, \sigma = 0.23°$) commensurable to a *9-Points* calibration ($\mu = 0.82°, \sigma = 0.15°$), well within physiological values, and calibration can be performed in ≈ 10 s, thus increasing gaze-based HCI usability. Future work includes developing and integrating methods to assess gaze estimation validity, compensating for calibration drift [36], and extending the proposed approach to 3D gaze estimations based, for instance, on vergence information [28].

REFERENCES

1. Javier San Agustin, Henrik H. T. Skovsgaard, Emilie Møllenbach, Maria Barret, Martin Tall, Dan Witzner Hansen, and John Paulin Hansen. 2010. Evaluation of a low-cost open-source gaze tracker. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications, ETRA 2010, Austin, Texas, USA, March 22-24, 2010*. 77–80. DOI: http://dx.doi.org/10.1145/1743666.1743685

2. Jason S. Babcock and Jeff B. Pelz. 2004. Building a lightweight eyetracking headgear. In *Proceedings of the Eye Tracking Research & Application Symposium, ETRA 2004, San Antonio, Texas, USA, March 22-24, 2004*. 109–114. DOI:http://dx.doi.org/10.1145/968363.968386

3. Wolfgang Becker. 1989. The neurobiology of saccadic eye movements. Metrics. *Reviews of oculomotor research* 3 (1989), 13.

4. Sacha Bernet, Christophe Cudel, Damien Lefloch, and Michel Basset. 2013. Autocalibration-based partioning relationship and parallax relation for head-mounted eye trackers. *Mach. Vis. Appl.* 24, 2 (2013), 393–406. DOI: http://dx.doi.org/10.1007/s00138-012-0427-3

5. Kamran Binaee, Gabriel J. Diaz, Jeff B. Pelz, and Flip Phillips. 2016. Binocular eye tracking calibration during a virtual ball catching task using head mounted display. In *Proceedings of the ACM Symposium on Applied Perception, SAP 2016, Anaheim, California, USA, July 22-23, 2016*. 15–18. DOI: http://dx.doi.org/10.1145/2931002.2931020

6. Lilian Calvet, Pierre Gurdjos, Carsten Griwodz, and Simone Gasparini. 2016. Detection and Accurate Localization of Circular Fiducials under Highly Challenging Conditions. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. 562–570. DOI: http://dx.doi.org/10.1109/CVPR.2016.67
7. Juan J. Cerrolaza, Arantxa Villanueva, and Rafael Cabeza. 2012. Study of Polynomial Mapping Functions in Video-Oculography Eye Trackers. *ACM Trans. Comput.-Hum. Interact.* 19, 2 (2012), 10. DOI: http://dx.doi.org/10.1145/2240156.2240158
8. Ergoneers. 2016. Dikablis Glasses. http://www.ergoneers.com/en/hardware/eye-tracking/. (2016).
9. Kai Essig, Marc Pomplun, and Helge J. Ritter. 2006. A neural network for 3D gaze recording with binocular eye trackers. *IJPEDS* 21, 2 (2006), 79–95. DOI: http://dx.doi.org/10.1080/17445760500354440
10. Karen M Evans, Robert A Jacobs, John A Tarduno, and Jeff B Pelz. 2012. Collecting and analyzing eye tracking data in outdoor environments. *Journal of Eye Movement Research* 5, 2 (2012), 6.
11. Wolfgang Fuhl, Thiago Santini, David Geisler, Thomas Kübler, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2016a. Eyes Wide Open? Eyelid Location and Eye Aperture Estimation for Pervasive Eye Tracking in Real-world Scenarios. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct (UbiComp '16)*. ACM, New York, NY, USA, 1656–1665. DOI: http://dx.doi.org/10.1145/2968219.2968334
12. Wolfgang Fuhl, Thiago C. Santini, Thomas Kübler, and Enkelejda Kasneci. 2016b. ElSe: Ellipse Selection for Robust Pupil Detection in Real-world Environments. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications (ETRA '16)*. ACM, New York, NY, USA, 123–130. DOI: http://dx.doi.org/10.1145/2857491.2857505
13. Wolfgang Fuhl, Marc Tonsen, Andreas Bulling, and Enkelejda Kasneci. 2016c. Pupil detection for head-mounted eye tracking in the wild: An evaluation of the state of the art. *Springer Machine Vision and Applications* (2016), 1–14. DOI: http://dx.doi.org/10.1007/s00138-016-0776-4
14. S. Garrido-Jurado, R. Mu noz Salinas, F.J. Madrid-Cuevas, and M.J. Marín-Jiménez. 2014. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition* 47, 6 (2014), 2280 – 2292. DOI: http://dx.doi.org/10.1016/j.patcog.2014.01.005
15. SensoMotoric Instruments GmbH. 2016. http://www.eyetracking-glasses.com/products/eye-tracking-glasses-2-wireless/technology/. (2016). Accessed: 16-09-07.
16. Dan Witzner Hansen and Qiang Ji. 2010. In the Eye of the Beholder: A Survey of Models for Eyes and Gaze. *IEEE Trans. Pattern Anal. Mach. Intell.* 32, 3 (2010), 478–500. DOI:http://dx.doi.org/10.1109/TPAMI.2009.30
17. Kenneth Holmqvist, Marcus Nyström, Richard Andersson, Richard Dewhurst, Halszka Jarodzka, and Joost Van de Weijer. 2011. *Eye tracking: A comprehensive guide to methods and measures*. Oxford University Press.
18. 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. DOI: http://dx.doi.org/10.3758/BF03195487
19. Michael Xuelin Huang, Tiffany C. K. Kwok, Grace Ngai, Stephen C. F. 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, San Jose, CA, USA, May 7-12, 2016*. 5169–5179. DOI: http://dx.doi.org/10.1145/2858036.2858404
20. Chris L Kleinke. 1986. Gaze and eye contact: a research review. *Psychological bulletin* 100, 1 (1986), 78.
21. Susan M. Kolakowski and Jeff B. Pelz. 2006. Compensating for eye tracker camera movement. In *Proceedings of the Eye Tracking Research & Application Symposium, ETRA 2006, San Diego, California, USA, March 27-29, 2006*. 79–85. DOI: http://dx.doi.org/10.1145/1117309.1117348
22. Thomas C. Kübler, Tobias Rittig, Enkelejda Kasneci, Judith Ungewiss, and Christina Krauss. 2016. Rendering refraction and reflection of eyeglasses for synthetic eye tracker images. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications, ETRA 2016, Charleston, SC, USA, March 14-17, 2016*. 143–146. DOI: http://dx.doi.org/10.1145/2857491.2857494
23. Pupil Labs. 2016. https://pupil-labs.com/. (2016). Accessed: 16-09-07.
24. Christian Lander, Frederic Kerber, Thorsten Rauber, and Antonio Krüger. 2016. A time-efficient re-calibration algorithm for improved long-term accuracy of head-worn eye trackers. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications, ETRA 2016, Charleston, SC, USA, March 14-17, 2016*. 213–216. DOI: http://dx.doi.org/10.1145/2857491.2857513
25. Dongheng Li, Jason S. Babcock, and Derrick J. Parkhurst. 2006. openEyes: a low-cost head-mounted eye-tracking solution. In *Proceedings of the Eye Tracking Research & Application Symposium, ETRA 2006, San Diego, California, USA, March 27-29, 2006*. 95–100. DOI: http://dx.doi.org/10.1145/1117309.1117350

26. Päivi Majaranta and Andreas Bulling. 2014. *Eye Tracking and Eye-Based Human–Computer Interaction*. Springer London, London, 39–65. DOI: http://dx.doi.org/10.1007/978-1-4471-6392-3_3

27. Craig H Meyer, Adrian G Lasker, and David A Robinson. 1985. The upper limit of human smooth pursuit velocity. *Vision research* 25, 4 (1985), 561–563.

28. Esteban Gutierrez Mlot, Hamed Bahmani, Siegfried Wahl, and Enkelejda Kasneci. 2016. 3D Gaze Estimation using Eye Vergence. In *Proceedings of the 9th International Joint Conference on Biomedical Engineering Systems and Technologies*. 125–131. DOI: http://dx.doi.org/10.5220/0005821201250131

29. Carlos Hitoshi Morimoto and Marcio R. M. Mimica. 2005. Eye gaze tracking techniques for interactive applications. *Computer Vision and Image Understanding* 98, 1 (2005), 4–24. DOI: http://dx.doi.org/10.1016/j.cviu.2004.07.010

30. M Nyström, R Andersson, K Holmqvist, and J Van de Weijer. 2011. Participants know best–the influence of calibration method and eye physiology on eye tracking data quality. *Journal of Neuroscience Methods* (2011), 1–26.

31. Marcus Nyström, Ignace Hooge, and Kenneth Holmqvist. 2013. Post-saccadic oscillations in eye movement data recorded with pupil-based eye trackers reflect motion of the pupil inside the iris. *Vision research* 92 (2013), 59–66.

32. Ken Pfeuffer, Mélodie Vidal, Jayson Turner, Andreas Bulling, and Hans Gellersen. 2013. Pursuit calibration: making gaze calibration less tedious and more flexible. In *The 26th Annual ACM Symposium on User Interface Software and Technology, UIST'13, St. Andrews, United Kingdom, October 8-11, 2013*. 261–270. DOI: http://dx.doi.org/10.1145/2501988.2501998

33. Hayes S Raffle and Chia-Jean Wang. 2015. Heads up display. (April 7 2015). US Patent 9,001,030.

34. Thiago Santini, Wolfgang Fuhl, David Geisler, and Enkelejda Kasneci. 2017. EyeRecToo: Open-Source Software for Real-Time Pervasive Head-Mounted Eye-Tracking. In *Proceedings of the 12th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2017) - To appear*.

35. Thiago Santini, Wolfgang Fuhl, Thomas C. Kübler, and Enkelejda Kasneci. 2016. Bayesian identification of fixations, saccades, and smooth pursuits. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications, ETRA 2016, Charleston, SC, USA, March 14-17, 2016*. 163–170. DOI: http://dx.doi.org/10.1145/2857491.2857512

36. Yusuke Sugano and Andreas Bulling. 2015. Self-Calibrating Head-Mounted Eye Trackers Using Egocentric Visual Saliency. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, NY, USA, 363–372. DOI: http://dx.doi.org/10.1145/2807442.2807445

37. Lech Świrski and Neil A. Dodgson. 2013. A fully-automatic, temporal approach to single camera, glint-free 3D eye model fitting [Abstract]. In *Proceedings of ECEM 2013*.

38. Tobii Technology. 2016. http://www.tobiipro.com/product-listing/tobii-pro-glasses-2/. (2016). Accessed: 16-09-07.

39. Eye Tribe. 2016. https://theeyetribe.com. (2016). Accessed: 16-09-07.

40. Arantxa Villanueva and Rafael Cabeza. 2008. A Novel Gaze Estimation System With One Calibration Point. *IEEE Trans. Systems, Man, and Cybernetics, Part B* 38, 4 (2008), 1123–1138. DOI: http://dx.doi.org/10.1109/TSMCB.2008.926606

41. Lawrence H. Yu and Moshe Eizenman. 2004. A new methodology for determining point-of-gaze in head-mounted eye tracking systems. *IEEE Trans. Biomed. Engineering* 51, 10 (2004), 1765–1773. DOI: http://dx.doi.org/10.1109/TBME.2004.831523

42. Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. 2015. Appearance-based gaze estimation in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. 4511–4520. DOI: http://dx.doi.org/10.1109/CVPR.2015.7299081