

Smooth-i: Smart Re-Calibration Using Smooth Pursuit Eye Movements

Argenis Ramirez Gomez
Lancaster University
Lancaster, United Kingdom
a.ramirezgomez@lancaster.ac.uk

Hans Gellersen
Lancaster University
Lancaster, United Kingdom
h.gellersen@lancaster.ac.uk

ABSTRACT

Eye gaze for interaction is dependent on calibration. However, gaze calibration can deteriorate over time affecting the usability of the system. We propose to use motion matching of smooth pursuit eye movements and known motion on the display to determine when there is a drift in accuracy and use it as input for re-calibration. To explore this idea we developed *Smooth-i*, an algorithm that stores calibration points and updates them incrementally when inaccuracies are identified. To validate the accuracy of *Smooth-i*, we conducted a study with five participants and a remote eye tracker. A baseline calibration profile was used by all participants to test the accuracy of the *Smooth-i* re-calibration following interaction with moving targets. Results show that *Smooth-i* is able to manage re-calibration efficiently, updating the calibration profile only when inaccurate data samples are detected.

CCS CONCEPTS

• **Human-centered computing** → **Interaction techniques**;

KEYWORDS

Gaze Calibration, Smooth Pursuits, Gaze interaction, Eye movements, Eye tracking

ACM Reference Format:

Argenis Ramirez Gomez and Hans Gellersen. 2018. *Smooth-i: Smart Re-Calibration Using Smooth Pursuit Eye Movements*. In *ETRA '18: 2018 Symposium on Eye Tracking Research and Applications, June 14–17, 2018, Warsaw, Poland*. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/3204493.3204585>

1 INTRODUCTION

Gaze has become a compelling modality for interaction. Users are able to interact with static and moving content just by looking, which has enabled a wide range of interaction techniques. However, in order to successfully estimate where users look, eye trackers require careful calibration to individual users. Conventional approaches to calibration often hamper usability as they require users to engage in a rigid procedure of fixating on a sequence of points on the target display [Morimoto and Mimica 2005]. However,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ETRA '18, June 14–17, 2018, Warsaw, Poland

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-5706-7/18/06...\$15.00

<https://doi.org/10.1145/3204493.3204585>

techniques have been developed that enable more implicit forms of calibration, based on motion correlation of moving stimuli and smooth pursuit eye movement [Pfeuffer et al. 2013; Ramirez-Gomez and Gellersen 2017].

In this work we address the problem that gaze calibration can become inaccurate over time. This occurs when the position of the user relative to the eye tracker is changing, and affects the user experience. Input actions become less accurate and more prone to failure, necessitating re-calibration [Feit et al. 2017]. In the lab, it has been experimenters ensuring that lapses in gaze accuracy were detected and corrected. However, as eye trackers become deployed as consumer devices, the onus is on the user to detect deviations in accuracy and the need to re-calibrate. There are currently no automated methods for detecting when the gaze estimation becomes inaccurate, and re-calibration remains a manual task.

We propose *Smooth-i* as an approach for automatic detection of inaccuracy in estimated gaze input, and incremental re-calibration as corrective action. It has previously been shown that smooth pursuit eye movement can be used to detect reliably when a user is looking at moving object, on the basis of motion correlation [Esteves et al. 2015; Ramirez-Gomez and Gellersen 2017; Velloso et al. 2017; Vidal et al. 2013]. In *Smooth-i*, we adopt the technique as a tool for measuring the accuracy of gaze input. As and when a correlation of eye movement with a moving object is detected during interaction, the object's position is compared with the eye tracker's estimated gaze point. If the error exceeds an accuracy threshold, a re-calibration is triggered. This is based on calibration points that are incrementally updated based on motion matching between gaze and targets at known positions on the screen.

Our contributions are a description of the *Smooth-i* algorithm, an experiment evaluating how effective it is for detecting and correcting gaze inaccuracy, and a scenario-based evaluation of *Smooth-i*'s incremental re-calibration performance.

2 RELATED WORK

The user experience of gaze-based controls is dependent on the accuracy of the eye trackers' gaze estimates, and high levels of accuracy can only be achieved through careful calibration to the individual user [Morimoto and Mimica 2005; Nyström et al. 2013; Schnipke and Todd 2000].

Most commonly used gaze calibration tasks require participants to fixate on (between 4 and 9) known static points distributed evenly around the screen. This process is usually separate from the application, and constitutes undesirable configuration overhead for the user. It has been reported to be of poor usability, boring, tedious and tiring for the eyes [Morimoto and Mimica 2005; Schnipke and Todd 2000; Villanueva et al. 2004].

There is wide-ranging on improving the usability of calibration, by reducing the number of calibration points [Guestrin and Eizenman 2008; Villanueva and Cabeza 2008], gathering manual user input to calibrate [Sellen et al. 2017], using head movements to collect a larger array of calibration points [Santini et al. 2017], and use of moving targets to ease calibration [Kondou and Ebisawa 2008; O’regan 1978]. Others have proposed the creation of calibration games [Flatla et al. 2011; Renner et al. 2011], playful embedding of calibration in the application [Dorr et al. 2009; Pfeuffer et al. 2013], and implicit calibration based on interaction dynamics within the application [Ramirez-Gomez and Gellersen 2017; Sidenmark 2017].

Smooth-i specifically builds on smooth pursuits, first proposed by Pfeuffer et al. for calibration [2013]. The use of smooth pursuits leverage the natural attention of the eyes when they try to fixate on a target that is moving [Gegenfurtner 2016], and has become a compelling tool for interaction [Esteves et al. 2015; Špakov et al. 2016; Vidal et al. 2013]. Previous work based on the use of smooth pursuits shows that motion matching provides relevant information on when the users are attentive to the presented motion and where are they looking. This advantage has been leveraged to demonstrate that the creation of an integrated and invisible gaze calibration is possible within the application dynamics [Ramirez-Gomez and Gellersen 2017].

In order to validate the created calibration’s gaze accuracy, tests are performed in experimental environments [Ohno and Mukawa 2004]. Traditionally, gaze accuracy is also assessed in a separate application by performing a series of fixations at given points on the screen, similar to the calibration process. Nevertheless, when and how to assess the accuracy of the gaze point remains the user’s responsibility. Users need to first comprehend that there is a drift on the provided gaze accuracy and then decide to re-calibrate.

With Smooth-i we propose to solve the gap between the detection of inaccuracies and trigger and execution of the re-calibration process by using smooth pursuit eye movements, enabling it to be performed automatically.

3 SMOOTH-I DESIGN

Smooth-i approaches the detection of smooth pursuits in a novel way, by leveraging the detection of where users look to both estimate gaze accuracy and re-calibrate ‘on-demand’. The method is implicit and opportunistic in that it collects data points automatically when the user happens to follow a moving target with their gaze. While a correlation is detected, the point positions of the moving target on the screen are associated with gaze points provided by the eye tracker, in a continuous sampling process.

Figure 1 illustrates how Smooth-i is structured. First, the algorithm performs motion-matching as it can only determine where the user is looking when the user’s gaze matches the motion of a known target’s trajectory. Second, the collected paired points are evaluated in an accuracy check step. The distance between them is evaluated against a desired gaze accuracy in degrees of visual angle so as to only propagate forward those points that show inaccuracies of the estimated gaze. Finally, the new paired points are stored to be used for gaze re-calibration.

For re-calibration, Smooth-i maintains a store of calibration points that is incrementally updated. When a Smooth-i application

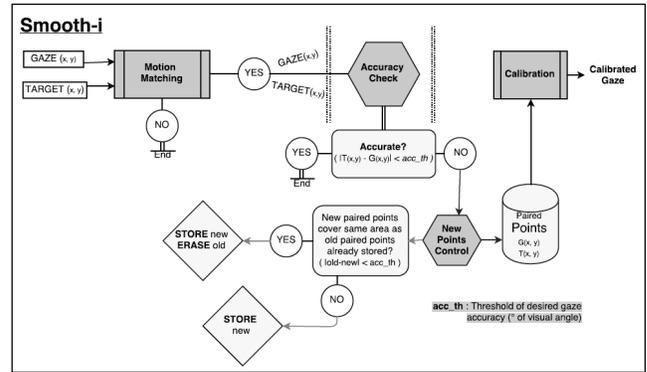


Figure 1: Smooth-i method. First, gaze and target moving points are evaluated through a motion matching procedure. All detected paired points are used to assess the accuracy. In case of inaccuracies, if there are no existing points in the area (used for re-calibration), new points are added. If there are old points in the same area, new points substitute them.

starts, the store might be empty or contain points from any prior calibration. In operation, the store is updated whenever inaccuracies are detected. These can be caused by either a lack of information in the area, in which case a new calibration point is added to the store, expanding the calibration profile. Inaccuracies can also be caused by previously stored calibration points that are causing the error. To address this, Smooth-i deletes any prior points from the store that are in proximity of the detected inaccuracy. Following the update, all the stored points are used to re-calibrate and create an accurate gaze to display mapping.

3.1 Implementation

In order to test Smooth-i, we implemented an application in Java, receiving gaze data from a Tobii EyeX remote eye tracker, collecting data at 60Hz, via a C# app. We used a 27” monitor (Resolution: 1920x1080, Aspect ratio: 16:9). The system shows up to 3 circles (20 pixels diameter) moving in a loop distributed around the screen for maximum spatial coverage in each scenario.

Smooth-i determines whether the user is following the displayed movement by using the *Pearson’s product-moment correlation* for motion matching with the threshold set at 0.9, as higher values result in fewer detection errors [Vidal et al. 2013].

Accuracy is evaluated in degrees of visual angle and computed by the difference in position between two paired points (gaze and target), and comparing it with a desired target accuracy. A predefined accuracy threshold of 1.5° was set for the evaluation of the method, as remote trackers’ average accuracy is often found to be larger than 1 degree [Feit et al. 2017; Lander et al. 2018].

Finally, we stored the selected calibration points and used them for re-calibration. Due to limitations at accessing Tobii’s raw data and restrictions to modify its calibration profile, we map the new calibration function on the given eye tracker’s estimation, producing a new calibrated gaze point based on Tobii’s gaze point. We used a second order polynomial geometric linear transformation for gaze to screen mapping (as a calibration function) [Morimoto and Mimica 2005].

3.2 Design requirements

Motion Matching for re-calibration: Application designers need to be aware that the method can only detect the accuracy in areas where motion occurs through motion matching. Smooth-i uses the collected points to create a calibration profile able to estimate (interpolate) the points within the area created by the stored points. Hence, it is not necessary to display motion in all the screen, but only where gaze pointing is going to be relevant.

Accuracy: The desired accuracy threshold needs to be set by the designers of the end application. How gaze input is going to be used in the final application would be determined by how detailed gaze information needs to be. For lower values, we approximate the estimation to "pixel perfect", whereas as the value increases, the resultant estimation would produce a limited gaze point only able to point at larger icons or regions of the screen.

4 EVALUATION

Smooth-i was evaluated in two experiments. First, in two steps, an evaluation of the accuracy detection compared to traditional methods, and an assessment of the acquired calibration's accuracy. Second, an evaluation of the performance of the method based on a series of user's interactions in 2 different scenarios. In each experiment, the researcher performed a different 16 points calibration based on fixations in order to start the system with a predefined inaccurate calibration profile, before each step.

4.1 Experiment on Accuracy Detection

In the first experiment, we integrated a 16 points accuracy test into Smooth-i. Users were required to fixate on each target while they were displayed. All accuracy targets (40 pixels diameter) were scattered evenly around the screen forming a 4x4 grid. Each target appeared on its own in a random order during a period of 2 seconds. We obtain the accuracy values by calculating the distance between the gaze point and the target point. We dismissed the information obtained during the first 0.8 seconds and the last 0.2 seconds related to the time participants' eyes need to travel to reach the target and anticipatory movements [Pfeuffer et al. 2013]. The result is computed by taking the median distance from all the gaze points to each target position, and later the mean between all of the targets. An accuracy of 1° of visual angle is considered good for remote eye trackers [Lander et al. 2018].

In the second part, after the researcher's calibration of the system, we tested how well Smooth-i can detect inaccuracies and correct them compared to a standard accuracy test. We deployed the system to show two target circles moving in a squared loop passing by all the points used in the standard accuracy test, so we could acquire close points and store the estimated accuracy with the motion matching detection. Those points detected during the pursuits near the 16 established points were considered as a section result.

We performed the evaluation with 5 participants (Age: $M = 25 \pm 2$, 4 male, 1 female, 2 wearing glasses). During the first part of the study, re-calibration was not considered, only motion matching. We asked each user to first perform a standard 16 points accuracy test. Later, we asked them to follow the motions displayed on the screen during two loops each. In the second part, we asked participants to follow the same procedure again with Smooth-i re-calibration

Table 1: Experiment 1 (Part 1): Mean Accuracy at the 16 points across the screen ($^\circ$ of visual angle)

Standard Test at the Start (Mean: 4.00 ± 1.08)

3.82 ± 1.22	4.27 ± 0.92	3.92 ± 1.03	3.57 ± 1.21
3.24 ± 0.64	3.98 ± 1.17	3.99 ± 0.95	4.37 ± 1.29
3.18 ± 1.09	4.01 ± 1.15	4.73 ± 1.38	4.38 ± 1.05
3.42 ± 1.02	4.21 ± 1.06	4.48 ± 0.94	4.41 ± 1.09

Smooth-i detection (Mean: 4.01 ± 1.23)

4.37 ± 1.56	3.45 ± 0.82	2.97 ± 0.90	2.85 ± 0.88
3.91 ± 0.68	4.40 ± 0.90	3.89 ± 1.32	4.41 ± 1.18
3.68 ± 1.31	4.48 ± 1.03	4.07 ± 1.61	3.87 ± 1.21
4.19 ± 1.40	4.33 ± 1.31	4.89 ± 1.42	4.36 ± 1.58

Table 2: Experiment 1 (Part 2): Evolution of the Mean Accuracy at the 16 points across the screen ($^\circ$ of visual angle)

Standard test at the Start (Mean: 2.64 ± 0.72)

2.57 ± 0.76	2.72 ± 0.69	2.54 ± 0.59	3.02 ± 0.87
1.87 ± 0.59	2.31 ± 0.78	2.82 ± 0.92	2.71 ± 0.91
2.26 ± 0.82	2.40 ± 0.54	2.7 ± 0.63	2.68 ± 0.97
2.83 ± 0.46	2.57 ± 0.40	3.24 ± 0.45	3.06 ± 0.81

Standard test after Smooth-i (Mean: 1.01 ± 0.48)

1.14 ± 0.20	1.10 ± 0.33	0.76 ± 0.68	0.75 ± 0.33
0.91 ± 0.43	0.90 ± 0.70	1.04 ± 0.90	1.10 ± 0.39
0.86 ± 0.26	0.95 ± 0.49	0.97 ± 0.66	1.10 ± 0.49
1.34 ± 0.34	0.79 ± 0.21	0.96 ± 0.37	1.46 ± 0.54

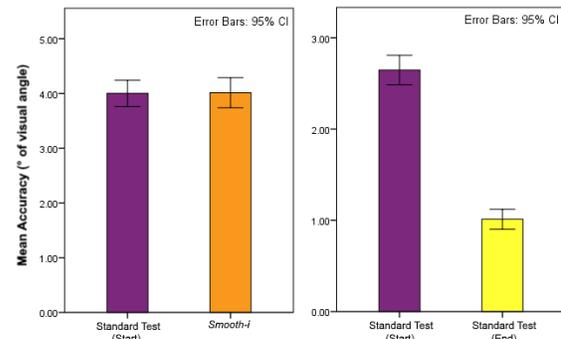


Figure 2: Mean Accuracy. (Left) Accuracy reported by the standard test ($4.00 \pm 1.08^\circ$) and Smooth-i ($4.01 \pm 1.23^\circ$). (Right) Accuracy at the start ($2.64 \pm 0.72^\circ$) and end of the study after Smooth-i ($1.01 \pm 0.48^\circ$).

being applied, followed by another standard accuracy test at the end. Results were calculated in degrees of visual angle.

Results: Table 1 and Table 2 show the mean screen accuracy at the 16 points scattered around the screen. Table 1 shows the differences between the standard accuracy test (at the start of the study) and the Smooth-i detection. Results show how both report inaccuracies, with a difference of $0.01 \pm 0.78^\circ$ (Figure 2, left). The mean of the absolute difference between both accuracies ($0.61 \pm 0.48^\circ$) show how pursuits report the re-calibrated gaze point to be further away from the target point than fixations.

Table 3: Results Experiments on Smooth-i Performance

	<i>T</i>	<i>User</i>	<i>Calibration Points</i>	<i>New / Erased</i>
<i>Turn taking</i>	1	1	28	12 / 0
	2	2	32	12 / 8
	3	1	35	11 / 8
	4	2	36	13 / 12
<i>Shared calibration</i>	1	1	20	5 / 1
	2	2	23	4 / 1
	3	3	34	11 / 0
	4	1	36	2 / 0
	5	2	42	7 / 1
	6	3	43	6 / 5
	7	1	47	15 / 11

We performed a *Wilcoxon signed-rank test* showing a not statistically significant difference between both methods for assessing accuracy ($Z = -.072$, $p = 0.943$). *Table 2* shows the mean accuracy from the accuracy tests performed during the study. It shows how Smooth-i improves accuracy from a mean of $2.64 \pm 0.72^\circ$ to $1.09 \pm 0.48^\circ$ (*Figure 2*, right), with little variance between users.

4.2 Experiment on Performance

In the second study, we evaluated how Smooth-i is able to update the calibration profile when different users use the same. We explored the performance of our proposed method in two case scenarios:

A) Turn taking: Two users using the same display at different times. Both use the system and then leave for two times.

B) Shared calibration Three users sharing the same display at different times. Each of them only needs access to one third of the screen. All three users come and leave two times. At the end, the first user comes back and uses all three parts of the screen.

The number of Calibration Points stored and used for later re-calibration, including the number of new points that were added or erased, were saved as a result after each user iteration. Before each case test starts, the system was given a 16 points calibration profile. Each of the users was asked to follow the motion of an assigned target during two loops each, and then leave.

For *Turn taking*, the system was configured so a single target performed a circular movement in the middle of the screen occupying most of the display. We tested how Smooth-i manages user changes.

In *Shared calibration*, the system was configured so that three targets (1 for each user) occupied a different section of the screen. We tested how Smooth-i manages different users sharing a calibration profile.

Results: *Table 3* shows the total amount of calibration points added after the given 16 points following each users' interaction (*T*) for each case study. Moreover, it presents how many new points were added during each stage and how many existing points from the calibration profile were erased to give space to a recently added ones. In *Turn taking*, we can observe how user 2 is updating the profile from user 1 with their gaze points, substituting 8 and 12 points. In *Shared calibration*, we can observe how after each user only interacted with their area, user 1 takes over the other's sections, substituting 11 points and adding 4. These results show that corrections and updates of the calibration profile occurred throughout the study.

5 DISCUSSION

Smooth Pursuit eye movements' attentive behavior when motion is presented enabled new calibration methods that are reported as more flexible and less tedious.

Leveraging the eyes' ability to follow movement, motion matching methods are able to identify when the user is looking at a moving target. Smooth-i identifies this behavior and uses it for accuracy assessment rather than a direct re-calibration. According to the detected accuracy, our proposed method decides whether re-calibration is needed, and it manages the calibration profile by erasing and adding points.

Results in *Table 1* show how the detection of accuracy differs from fixation and pursuits based methods. The mean accuracy for the screen is similar (4° and 4.01°), and both methods are reported to be statistically equal. However, the detection based on pursuits is reported to score greater inaccuracies (absolute difference mean $0.61 \pm 0.48^\circ$) than the standard test. We believe this is an effect of the difference between the eye movements behavior when selecting targets [Lohr and Komogortsev 2017], and it can be related to the phase shift (lag) of pursuits during this movement [Holmqvist et al. 2011]. Nevertheless, *Figure 2* shows how the proposed method is able to detect a drift in gaze accuracy for later correction as the standard method would do.

Moreover, the results shown in *Table 2* and *Figure 2* show that Smooth-i is a competent method for re-calibration. It was able to improve a detected inaccuracy of 2.64° to 1.01° . Accuracies of 1° are considered good results [Lander et al. 2018], even if greater results are often reported in controlled environments for remote eye trackers [Feit et al. 2017; Nyström et al. 2013].

Further, different scenarios were tested to assess Smooth-i's management of calibration points. *Table 3* shows how, on each user's interaction with the system, the profile is being updated as one user's points take over the profile. On the other hand, the second scenario presents further evidence that samples for 3 different users could be collected to be used in different parts of the screen within the same profile. Our results suggest this approach might broaden opportunities to use gaze in shared displays and multi-user systems.

Moreover, Smooth-i has the potential to be integrated inside applications containing moving content. Examples range from screen savers or locked screens which would require explicit interaction with moving content [Pfeuffer et al. 2013], games [Ramirez-Gomez and Gellersen 2017; Vidal et al. 2013] that could be extended with gaze interaction, or even infographics (motion graphics) videos with moving animations that would allow re-calibrating gaze implicitly.

6 CONCLUSION

We proposed Smooth-i as a novel method for automatic re-calibration using smooth pursuit eye movements. The method presents a new approach on the detection of smooth pursuits to assess the accuracy of the gaze point and automatically re-calibrate when required. Results show how the system is able to manage re-calibration only when inaccuracies are detected by adding new samples and erasing outdated points. Smooth-i leads towards more usable and intelligent strategies for re-calibration, while maximizing its potential for multiuser systems or shared displays.

REFERENCES

- Michael Dorr, Laura Pomarjansch, and Erhardt Barth. 2009. Gaze beats mouse: A case study on a gaze-controlled breakout. *Psychology Journal* 7, 2 (2009), 197–211.
- Augusto Esteves, Eduardo Velloso, Andreas Bulling, and Hans Gellersen. 2015. Orbits: Gaze Interaction for Smart Watches using Smooth Pursuit Eye Movements. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. ACM, 457–466.
- Anna Maria Feit, Shane Williams, Arturo Toledo, Ann Paradiso, Harish Kulkarni, Shaun Kane, and Meredith Ringel Morris. 2017. Toward Everyday Gaze Input: Accuracy and Precision of Eye Tracking and Implications for Design. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 1118–1130. <https://doi.org/10.1145/3025453.3025599>
- David R Flatla, Carl Gutwin, Lennart E Nacke, Scott Bateman, and Regan L Mandryk. 2011. Calibration games: making calibration tasks enjoyable by adding motivating game elements. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*. ACM, 403–412.
- Karl R Gegenfurtner. 2016. The interaction between vision and eye movements. *Perception* 45, 12 (2016), 1333–1357.
- Elias Daniel Guestrin and Moshe Eizenman. 2008. Remote point-of-gaze estimation requiring a single-point calibration for applications with infants. In *Proceedings of the 2008 symposium on Eye tracking research & applications*. ACM, 267–274.
- 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*. OUP Oxford.
- Yuki Kondou and Yoshinobu Ebisawa. 2008. Easy eye-gaze calibration using a moving visual target in the head-free remote eye-gaze detection system. In *Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2008. VECIMS 2008. IEEE Conference on*. IEEE, 145–150.
- Christian Lander, Markus löchtfeld, and Antonio Krüger. 2018. hEYEBrid: A Hybrid Approach for Mobile Calibration-free Gaze Estimation. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 4, Article 149 (Jan. 2018), 29 pages. <https://doi.org/10.1145/3161166>
- Dillon James Lohr and Oleg V Komogortsev. 2017. A Comparison of Smooth Pursuit- and Dwell-based Selection at Multiple Levels of Spatial Accuracy. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. ACM, 2760–2766.
- Carlos H Morimoto and Marcio RM Mimica. 2005. Eye gaze tracking techniques for interactive applications. *Computer Vision and Image Understanding* 98, 1 (2005), 4–24.
- Marcus Nyström, Richard Andersson, Kenneth Holmqvist, and Joost Van De Weijer. 2013. The influence of calibration method and eye physiology on eyetracking data quality. *Behavior research methods* 45, 1 (2013), 272–288.
- 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.
- Kevin O'regan. 1978. A new horizontal eye movement calibration method: Subject-controlled "smooth pursuit" and "zero drift". *Behavior Research Methods & Instrumentation* 10, 3 (1978), 393–397.
- Ken Pfeuffer, Melodie Vidal, Jayson Turner, Andreas Bulling, and Hans Gellersen. 2013. Pursuit calibration: Making gaze calibration less tedious and more flexible. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM, 261–270.
- Argenis Ramirez-Gomez and Hans Gellersen. 2017. GazeBall: Leveraging Natural Gaze Behavior for Continuous Re-calibration in Gameplay. *Journal of Eye Movement Research* 10, 6 (2017). <https://bop.unibe.ch/index.php/JEMR/article/view/4182>
- Patrick Renner, Nico Lüdike, Jens Wittrowski, and Thies Pfeiffer. 2011. Towards Continuous Gaze-Based Interaction in 3D Environments—Unobtrusive Calibration and Accuracy Monitoring. In *GI VRAR workshop, Shaker Verlag*.
- Thiago Santini, Wolfgang Fuhl, and Enkelejda Kasneci. 2017. CalibMe: Fast and Un-supervised Eye Tracker Calibration for Gaze-Based Pervasive Human-Computer Interaction. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, 2594–2605.
- Susan K Schnipke and Marc W Todd. 2000. Trials and tribulations of using an eye-tracking system. In *CHI'00 Extended Abstracts on Human Factors in Computing Systems*. ACM, 273–274.
- Abigail Jane Sellen, Adam Gary Emfield, Arridhana Ciptadi, Kenneth Robert Woodberry, and Andrew Blake. 2017. Gaze calibration. (Aug. 8 2017). US Patent 9,727,135.
- Ludwig Sidenmark. 2017. Immersive Eye Tracking Calibration in Virtual Reality Using Interactions with In-game Objects. (2017).
- Oleg Špakov, Poika Isokoski, Jari Kangas, Deepak Akkil, and Päivi Majaranta. 2016. PursuitAdjuster: an exploration into the design space of smooth pursuit-based widgets. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*. ACM, 287–290.
- Eduardo Velloso, Marcus Carter, Joshua Newn, Augusto Esteves, Christopher Clarke, and Hans Gellersen. 2017. Motion Correlation: Selecting Objects by Matching Their Movement. *ACM Trans. Comput.-Hum. Interact.* 24, 3, Article 22 (April 2017), 35 pages. <https://doi.org/10.1145/3064937>
- Mélie Vidal, Andreas Bulling, and Hans Gellersen. 2013. Pursuits: spontaneous interaction with displays based on smooth pursuit eye movement and moving targets. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*. ACM, 439–448.
- Arantxa Villanueva and Rafael Cabeza. 2008. A novel gaze estimation system with one calibration point. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* 38, 4 (2008), 1123–1138.
- Arantxa Villanueva, Rafael Cabeza, and Sonia Porta. 2004. Eye tracking system model with easy calibration. In *Proceedings of the 2004 symposium on Eye tracking research & applications*. ACM, 55–55.