# Advantages of Eye-Gaze over Head-Gaze-Based Selection in Virtual and Augmented Reality under Varying Field of Views

Jonas Blattgerste, Patrick Renner & Thies Pfeiffer
CITEC - Cluster of Excellence Cognitive Interaction Technology
Bielefeld University
{jblattgerste,prenner,tpfeiffer}@techfak.uni-bielefeld.de

## ABSTRACT

The current best practice for hands-free selection using Virtual and Augmented Reality (VR/AR) head-mounted displays is to use head-gaze for aiming and dwell-time or clicking for triggering the selection. There is an observable trend for new VR and AR devices to come with integrated eye-tracking units to improve rendering, to provide means for attention analysis or for social interactions. Eye-gaze has been successfully used for human-computer interaction in other domains, primarily on desktop computers. In VR/AR systems, aiming via eye-gaze could be significantly faster and less exhausting than via head-gaze.

To evaluate benefits of eye-gaze-based interaction methods in VR and AR, we compared aiming via head-gaze and aiming via eye-gaze. We show that eye-gaze outperforms head-gaze in terms of speed, task load, required head movement and user preference. We furthermore show that the advantages of eye-gaze further increase with larger FOV sizes.

## CCS CONCEPTS

• **Human-centered computing** → **Mixed / augmented reality**; **Virtual reality**; **Empirical studies in HCI**; *Pointing*;

## KEYWORDS

Augmented Reality; Virtual Reality, Assistance Systems; Head-Mounted Displays, Eye-Tracking, Field of View, Human Computer Interaction.

**ACM Reference Format:**
Jonas Blattgerste, Patrick Renner & Thies Pfeiffer. 2018. Advantages of Eye-Gaze over Head-Gaze-Based Selection in Virtual and Augmented Reality under Varying Field of Views. In *COGAIN '18: Workshop on Communication by Gaze Interaction, June 14–17, 2018, Warsaw, Poland*, Jennifer B. Sartor, Theo D'Hondt, and Wolfgang De Meuter (Eds.). ACM, New York, NY, USA, Article 4, 9 pages. https://doi.org/10.1145/3206343.3206349


*COGAIN '18, June 14–17, 2018, Warsaw, Poland*

ACM ISBN 978-1-4503-5790-6/18/06…$15.00
https://doi.org/10.1145/3206343.3206349

## 1 INTRODUCTION

Despite of huge developments in the last years, eye-tracking technology is still mainly used in research and did not yet reach consumers to substantial extend. However, for enabling rendering on upcoming high resolution displays (4k to 8k and above) with high performance, eye tracking could be a game changer, as foveated rendering has been shown to significantly reduce the computational requirements for rendering of 3D content [8, 19].

When eye-tracking technology becomes available on a broader scope, it can not only be used for enhancing performance, but also for user interaction. This arises the question of possible benefits of using eye tracking compared to classic interaction techniques for head-mounted displays (HMDs). In the domain of interaction with immersive user interfaces (UIs) in virtual reality (VR), a common selection technique in mobile systems, such as Google Cardboard, Samsung GearVR or Oculus Go, is using head-gaze for aiming at the UI elements: With this technique the head has to be moved so that a virtual cursor in the middle of the display overlaps with the desired UI element. With eye-tracking technology, head-gaze could be replaced by eye-gaze, which should reduce the requirements of head movements and speed up the time for aiming at the desired user interface element. The selection then has to be completed by a trigger event, which, e.g., can be realized either by clicking on a controller or a dwelling threshold to avoid the midas touch problem [14]. The nature and kind of the triggering event is beyond the scope of this paper.

While in VR applications controllers may be used for aiming and triggering, augmented reality (AR) HMDs are often used in tasks such as industrial assembly or maintenance. Here, it is essential to have both hands free for the task. Thus, mobile AR systems would have an increased benefit from eye-gaze-based aiming [23]. AR systems of today, however, have a much lower field of view (FOV) compared to VR systems. This requires head movements to bring virtual content into view that does not fit on the display. As a consequence, the width of the FOV is an independent variable in our study.

The paper is structured as follows: First, related work on selection using eye-gaze-based techniques in VR and AR is discussed. Then, previous research regarding the use of simulated AR is presented. This is in particular necessary, as no high-quality eye-tracking for AR glasses is currently available and we thus use VR simulation with an eye-tracking HMD as proxy for our analysis. Furthermore, the realized interaction methods, as well as the UI elements used for evaluation are described in detail. After that, the study methodology and results are presented and discussed.

## 2 RELATED WORK

Eye tracking has been recognized as a promising interaction technology for mobile AR systems (see e.g. [10]) and several approaches have demonstrated its applicability for a diverse range of applications [18, 28, 29, 35]. However, today we still do not find any commercially available AR system with eye tracking build-in. This is different for VR headsets, where established eye tracking system builders, such as SMI [13] or Tobii [1], offer to integrate eye tracking into consumer HMDs for immersive VR. In addition to that, start-up companies offer VR headsets with build-in eye tracking, such as FOVE [12] or Looxid Labs [20].

### 2.1 Object Selection

An important user action in AR and VR systems is object selection. The interaction scenario targeted at with this paper is an immersive command-and-control-like scenario with a focus on object selection, similar to a standard desktop scenario in which mouse movements are used for aiming and a button press for triggering the selection. The application of eye-gaze in the desktop scenario has been described by Jacob [14] but there is also already research done in the domain of VR. Cournia et al. [6] compared eye-gaze and hand-based pointing and found that hand-based pointing outperformed eye-gaze in terms of task-completion time. In fast-paced scenarios (e.g. VR games) Hülsmann et al. [11] found that controller-based pointing outperforms eye-gaze in terms of accuracy but not task-completion times or immersion of the user. Tanriverdi et al. [34] found that eye-gaze was faster than pointing in virtual environments and that this effect increases for distant objects. Sibert et al. [32] compared eye-gaze to mouse selection for virtual environments and found that eye-gaze performed better in terms of task-completion times.

Finally, Qian et al. [30] compared eye-gaze, head-gaze, and a combination of eye- and head-gaze in VR using the FOVE HMD [12]. They found that, contrary to their expectations, head-gaze outperformed both, the eye- and the eye/head-gaze combination in terms of errors made, task completion time and subjective ratings. However, they report that those results may have been caused by accuracy problems with the eye tracker and, furthermore, that qualitative feedback indicated that some participants would still, despite those problems, prefer eye tracking and perceived it as more comfortable to use. In our study, we employ a high-resolution eye-tracking system by SMI to overcome the accuracy problems of this previous research. While Qian et al. investigated three conditions, we believe that their eye-gaze condition with a fixed head is not of interest for any real-life application. We refer to the condition Qian et al. reported as eye/head-gaze combination as eye-gaze in this paper and will no further explore the condition they described as eye-gaze only.

Implicit interactions or natural user interfaces, in which gaze can be used to disambiguate object references made via speech or gestures, are not addressed, but the reported findings may also apply to them.

*Object Selection is not Typing:* An area of human-computer interaction in which eye tracking has a prominent appearance is assistive computing and in particular eye-gaze-based typing (see e.g. [9, 22]). It has to be noted, that the work presented here does make use of a typical keyboard layout to realize multiple selection targets. However, the focus of the paper is not on writing text, but on a more general use of eye-gaze for selection. The competence of producing text on a keyboard has a large variability, depending in particular on the individual experience in typing on a keyboard, typing using a particular layout of a keyboard and the familiarity with the text. As a result, typing a text includes a learning component, which an experimental design has to take into account, e.g. by inviting participants multiple times. In contrast, the experimental design in this work is based on a basic stimulus and response paradigm, in which the target location itself is highlighted and the user simply has to react by triggering a selection on the target. The results, then, are not directly comparable to those reported for eye-gaze-based typing. However, we would argue that writing only adds a different kind of search phase before the selection process, which would be equivalent for head-gaze and eye-gaze-based typing and thus the differences in timing between the two techniques reported here should transfer to writing as well.

### 2.2 Simulated Augmented Reality

For the evaluation of the effect of different fields of views on performance of AR systems, a VR simulation of AR is used [15, 31, 37]. This has several reasons: first, there are currently no high-resolution eye-tracking systems available for AR. Second, this way the same device can be used for VR and AR, which enables the comparison between the FOV conditions. And, third, conducting an experiment using AR requires more effort than using VR (see e.g. [16, 17, 24]), as extreme care has to be taken to create a stable outside environment, in particular regarding the lighting conditions. VR simulations of AR systems can be used to virtualize the AR device [15, 31, 37] and to virtualize the interaction set-up and therefore provide a precise specification of the experiment.

VR simulation of AR has already been used to successfully identify task relevant-parameters of AR system designs (e.g. [25, 33]). Arthur [2], for example, evaluated different FOVs using simulated AR glasses and showed that a limited FOV reduced task performance in the examined scenario.

This paper builds upon the reported experiences and extends them in various ways: The simulation covers the display of a Microsoft HoloLens [26] (36°), an OmniVision/ASTRI [3] (60°) and a Meta 2 [5] (90°). However, the main difference to prior work is the focus on the design of the AR user interface (not the system) and thus on prototyping AR interaction in VR.

The ultimate goal would be the development of a user model for object selection in AR and VR, similar to Fitts' law [7], which describes the time required to aim at a target in terms of the target width (aka required precision) and the distance to the target (task index of difficulty) [21]. That this covers eye movements in desktop-based HCI as well has already been shown [27, 38]. The situation in VR and especially AR with limited FOVs, however, is slightly more difficult than for the desktop. The UI may be partly invisible and cover much larger areas, up to 360° around the user. This will almost always entail eye movements, head movements and ultimately body movements with increasing distance. The presented work, as a first step, addresses situations with FOVs that require eye and head movements.

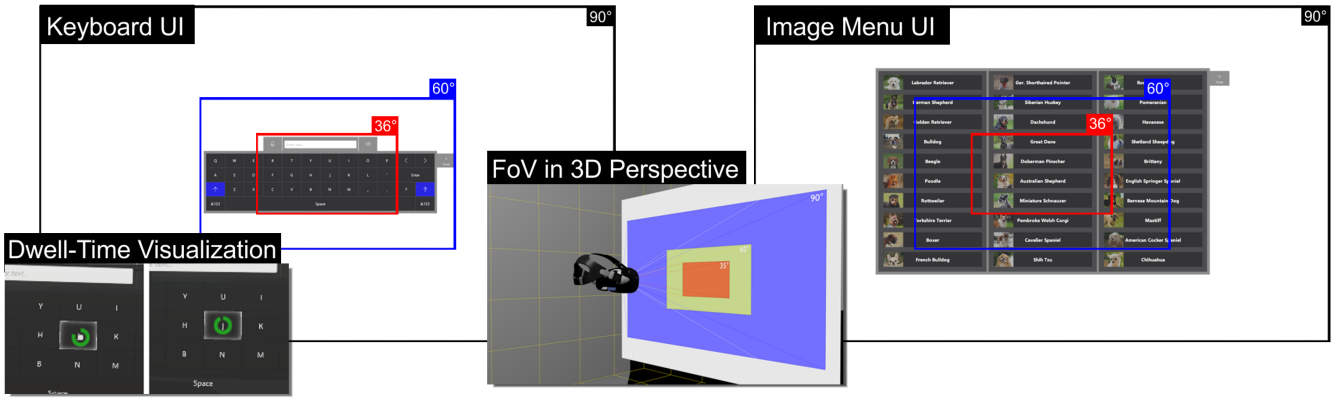


**Figure 1: *Keyboard UI*: The keyboard was fully visible in the 90° FOV and 60° FOV condition, but there only with the head oriented straight forward. It was partly visible in the 36° FOV condition. *Image Menu UI*: The image menu was only fully visible in the 90° FOV condition. The 60° FOV condition covered the width of 2.5 columns and the 36° FOV condition of one column. *Dwell-Time Visualization*: around the gaze cursor for head-gaze interaction (left) and central on UI element for eye-gaze interaction (right). *FoV in 3D Perspective*: A virtual reality HMD (110 ° FOV) with eye tracking was used to simulate HMDs with 36, 60, and 90 °.**

## 3 IMPLEMENTATION

To validly compare eye-gaze and head-gaze based interaction in a controllable environment, the interaction scenario of our evaluation was created in VR and simulated an AR display with configurable FOV sizes in front of the user's head. With this approach, all presentations could be realized with the same programming system and the same HMD (HTC Vive) and eye tracking device (SMI) for all conditions. This approach makes the results easily reproducible and introduces the opportunity to compare the interaction methods on different FOV sizes up to the FOV provided by the HMD (110°).

In the simulation, two interaction methods were implemented only differing in the modality used for aiming as well as two kinds of interactable UI elements. In addition to that, three different FOV sizes were simulated.

### 3.1 Object Selection

The process of object selection can be divided into the phase of *aiming* at the intended target followed by a *triggering* of the selection. Often *feedback* is provided during either or both of the two phases (e.g. by highlighting or a progress animation, or by providing auditive feedback signals). In this work we are focusing on the aiming phase and the effect of using either of the two modalities head-gaze or eye-gaze to control this process.

*Aiming with Histogram-based Filtering:* A ray was cast into the scene and tested for intersections with interactable elements. A histogram-based filter, motivated by [34], was used to reduce eye tracking instabilities and to prohibit the loss of targets during blinking. The histogram window was set to 30 frames (0.333 s), a parameter derived from a non-representative pilot study, in which this setting covered the large majority of blinks without creating to much noticeable latency. The one UI element that was hit for more than half of the frames covered by the histogram was used for the dwell-time triggering. This is depicted in Figure 2.

*Triggering using Dwell Time:* The dwell-timer was started after 0.3 s of interaction, including the length of the histogram. The dwell time threshold was set to 1 s, resulting in an overall dwell-time of 1.3 s. This setting was reported to feel the most responsive and natural for both interaction methods while not introducing false positives for participants in the pilot study. This study is about the differences in the aiming process and the triggering mechanism is kept constant for all conditions. Thus the focus was on robustness and low numbers of false positives as to not frustrate the users and get reliable results for the relevant part of interaction.

*Eye-Gaze Interaction Method:* The ray direction for aiming was defined by the binocular ray provided by the SMI eye tracker. Feedback was given on the progressing dwell time using a filling circle positioned on the center of the interactable object itself (see Figure 1, Dwell-Time Visualization, right picture). No gaze cursor was shown to the user in this condition.

*Head-Gaze Interaction Method:* The ray direction for aiming was defined by the viewing direction of the HMD. A feedback had to be given to the user, which was realized as a white dot projected in the center of the FOV. (see Figure 1, Dwell-Time Visualization, left picture) Feedback was also given on the progressing dwell time using a filling circle, but this time it was positioned around the white dot in the center of the FOV. Otherwise, the user would have had to focus on two locations (white dot and dwell time feedback) at the same time. The non-representative preliminary study indicated that this solution felt more natural for the participants.

### 3.2 The Interactable UI Elements

To compare the interaction methods in realistic scenarios, we implemented interfaces with two kinds of UI elements: The first being a 0.6 m × 0.17 m large replica of the keyboard that is used on the Microsoft HoloLens, with 0.038 m × 0.038 m sized buttons (see Figure 1). The second being a 0.85 m × 0.52 m large, menu-shaped UI

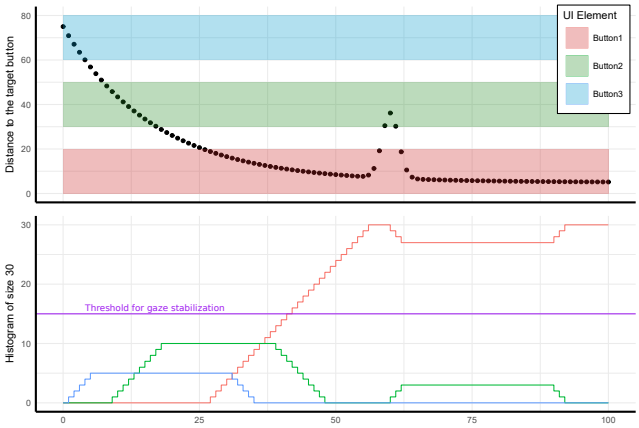


**Figure 2: The plot shows the effect of the algorithm used for object selection. The upper graph shows the movement of the eye (here only 1-dimensional) and the mapping to different GUI elements (colored ranges). The lower graph shows the entries in the histogram used to filter the areas of interest data (window size 30 frames). Only if half of the frames contain gaze points on a specific UI element, dwell-time tracking is started.**

element, with 0.25 m × 0.038 m sized menu-buttons, that is supposed to resemble UIs like menus that are already used in VR today (e.g. Samsung Gear VR menus) and may become standard on AR devices with increasing FOV sizes as well (see Figure 1).

For current AR UI implementations the best practice is to use tag-along implementations. Contrary to the standard sphere-based tag-along, the UI element was positioned with a fix distance of 0.7 meters in front of the user and adjusted to the height of the user once on startup to make the results comparable.

## 4 METHODOLOGY

The design was a within-subjects comparison, conducted as a repeated measures experiment with the independent variables FOV size (3), interface type (2) and interaction method (2), resulting in 2x2x3=12 conditions. The FOV sizes were (see Figure 1): 36° as small, 60° as medium and 90° as large. The dependent variables were the time to complete a full task, the average time to activate one element, the amount of errors made, the head and eye movement for each element and NASA (raw) TLX scores.

To prohibit possible systematic bias due to order effects (e.g. learning effects), we balanced the order of conditions using Latin square. We furthermore alternated the order of presentation of the keyboard and the menu task within the Latin squares.

After each task combination (keyboard and menu), responses were collected for the evaluation of the combination of FOV and interaction method. For the qualitative data and NASA (raw) TLX scores, we asked the participants to fill out questionnaires. All other quantitative data was collected on the device itself.

Participants conducted the experiment in a standing position in a virtual room that was modeled to resemble the original room the study took place in. The study is compliant with the ethical guidelines enforced by the ethical committee at our university.

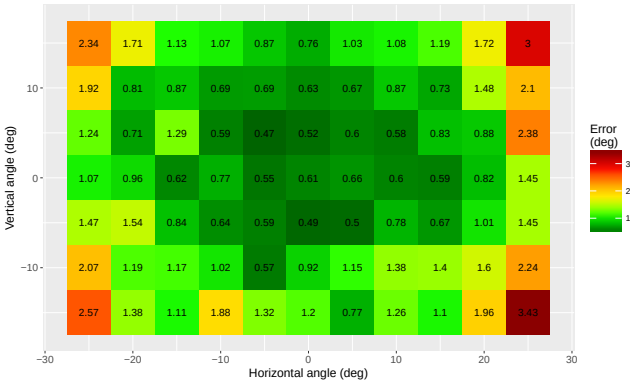


**Figure 3: The mean accuracy measurements for the 77 tested visual angles.**

### 4.1 Hardware

The AR simulation was implemented in Unity3D version 2017.1 for the virtual reality HMD HTC Vive. The Vive was connected to a computer with an Intel i7-6700K and a NVidia GeForce GTX 1080 to achieve a stable display frame rate of 90 Hz (maximum supported by HTC Vive).

Additionally, the HMD was equipped with the binocular 250 Hz "SMI HTC Vive Integration Scientific Premium" eye tracker with a reported accuracy of 0.2°. The accuracy in realistic usage was tested with 10 participants without debilities of sight. In a regular grid, we recorded data for visual angles within 50° horizontal and 30° vertical FOV. This approximately corresponds to the medium (60° diagonal) FOV. The measurements were taken in intervals of 5° both vertically and horizontally, so overall $11 * 7 = 77$ angles were measured. For each point, 200 measurements were recorded per participant. Figure 3 shows the measured accuracy for all tested angles. The average measured accuracy was 1.15° (SD=1.84). Within the 35° diagonal FOV of the HoloLens, the mean accuracy was 0.66° (SD=1.38). The best mean accuracy a participant reached was 0.72° (SD=1.2) overall and 0.39° (SD=1.09) within the HoloLens FOV.

### 4.2 Procedure

First, participants were asked to sign the declaration of consent and fill out a demographic questionnaire that also included questions about possibly impaired vision and previous experience in VR, AR and computer games. After that, they were given a short introduction to the HTC Vive and the conducted experiment and an introductory scene was started on the HMD to explain the AR display simulation, the two interaction methods and the UI elements. When participants reported they had understood the instructions, the eye-tracker was calibrated and the experiment was started.

Participants then started with one of the 12 conditions. Each condition contained 50 transitions between elements, which were gathered in one chunk. Of these 50 transitions, 50% were inside and 50% were outside of the smallest FOV. The order of the transitions was randomized to prohibit learning effects. It was ensured that the target elements were evenly distributed across all directions (all transition vectors added up to a zero vector). For each step, the target was highlighted in white. After completing the tasks on

both UI elements for a specific combination of FOV and interaction method, the participants were asked to take off the HMD and fill out the questionnaires. This was repeated for each combination.

After completing all conditions, participants were handed a final questionnaire that asked them which of the interaction methods they would prefer, why they would prefer it and if the size of the FOV had an impact on their decision. Furthermore, they were also asked for any additional feedback regarding the experiment.

### 4.3 Participants

We conducted the experiment with 24 participants that were aged between 19 and 32 (average = 23.54, SD = 3.04), 17 of the participants were female. While 12 participants had impaired vision, 10 of them completed the experiment without their glasses, the other two wore contact lenses. 11 participants reported previous experience with virtual reality, 10 reported previous experience with augmented reality and 16 reported experience with computer games. All of the participants were students of our university.

## 5 RESULTS

As objective measures we recorded the average time participants needed to perform an action on the current UI element, the average head movement needed and number of erroneous dwell timer activations. The results can be found in tables 1 and 2. Moreover, the perceived cognitive load was measured in form of a NASA (raw) TLX score. We asked for qualitative feedback and the user preference regarding the interaction methods.

As the measured data regarding task-completion-time, head movement and errors were skewed and thus violated the normality assumption for an ANOVA, they were preprocessed using the Aligned Rank Transform for non-parametric analysis [39]. The within-factor post-hoc analyses were done using pairwise Tukey-corrected Least Squares Means. Cross-factor pairwise comparisons were done using Wilcoxon signed-rank tests with Bonferroni-Holm correction.

### 5.1 Task Completion Times

When looking at the time participants needed to perform an action in form of one key activation on the keyboard UI element (see Figure 4, left), participants were the fastest using the eye-gaze method on the medium FOV with an average of 1.74 seconds, closely followed by the eye-gaze method on the large FOV. Participants were slowest while using the head-gaze method on the small FOV with an average of 2.1 seconds. While the time until first inspection is similar for eye gaze and head gaze, it changes with FOV size. The differences between task completion time using eye-gaze or head-gaze then result from the time between first inspection and triggering the UI element. The average performance increase for aiming (not taking into account the dwell time of 1.3 s for triggering) using eye-gaze instead of head-gaze was 239 ms or in other words, eye-gaze could reduce time-on-task by 31.8%.

Observing the time participants needed to perform an action in form of a button activation on the menu UI element (see Figure 4, right), the participants were fastest using the eye-gaze method on the large FOV size with an average of 2.04 seconds, followed by the head-gaze method on the large FOV. Using the head-gaze method

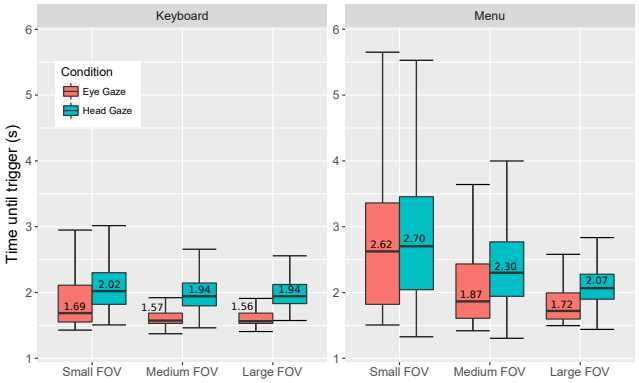


**Figure 4: The time participants needed to trigger an UI element.**

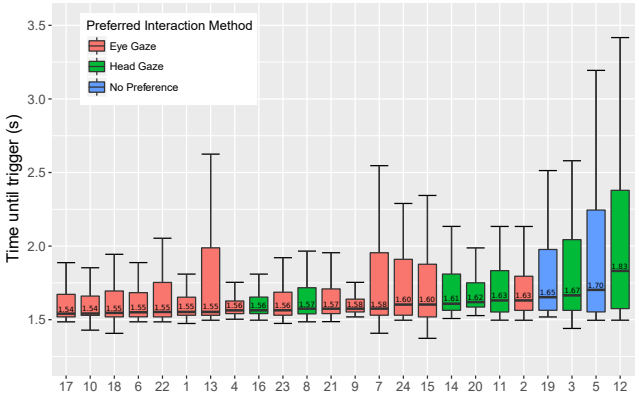


**Figure 5: The time each participant needed to trigger an UI element in the keyboard scenario. The majority of slower participants reported they preferred head gaze interaction or had no preference.**

on the small FOV they were the slowest with 2.86 seconds. In the menu scenario, on average using eye-gaze reduced aiming time by 138 ms (11,7%).

Figure 5 shows the individual performance of participants using eye-gaze in the keyboard scenario. Seven of the slower performing half of twelve participants reported they would prefer head-gaze over eye-gaze or had no preference. The fastest eight participants reported they preferred eye gaze interaction.

The ANOVA revealed significant main effects of interaction method and FOV size in the keyboard scenario as well as in the menu scenario ($p < .001$). Moreover, there is an interaction between both factors ($p < .001$). The post-hoc tests showed that in the keyboard condition, all differences in task completion time were highly significant ($p < .001$) except for the differences between large and medium FOV both for eye-gaze and head-gaze. For the menu scenario, the post-hoc tests showed that all differences were highly significant ($p < .001$).

**Table 1: Study results in the keyboard scenario (SD in brackets)**

| Condition | FOV Size | First Inspection (s) | Time until Trigger (s) | Time-on-task (s) | Head Movement (deg) | Errors |
|---|---|---|---|---|---|---|
| **Eye gaze** | small | 0.45 (0.48) | 1.49 (0.57) | 1.94 (0.72) | 11.11 (10.5) | 0.51 (1.28) |
| | medium | 0.28 (0.14) | 1.46 (0.54) | 1.74 (0.55) | 2.78 (4.88) | 0.29 (0.87) |
| | large | 0.27 (0.12) | 1.48 (0.66) | 1.75 (0.69) | 3.44 (6.12) | 0.29 (0.84) |
| **Head gaze** | small | 0.42 (0.44) | 1.68 (0.41) | 2.10 (0.4) | 17.50 (11.27) | 0.56 (0.80) |
| | medium | 0.33 (0.19) | 1.69 (0.31) | 2.02 (0.34) | 15.86 (9.68) | 0.40 (0.69) |
| | large | 0.29 (0.22) | 1.75 (0.46) | 2.03 (0.42) | 15.65 (9.74) | 0.44 (0.77) |

**Table 2: Study results in the menu scenario (SD in brackets)**

| Condition | FOV Size | First Inspection (s) | Time until Trigger (s) | Time-on-task (s) | Head Movement (deg) | Errors |
|---|---|---|---|---|---|---|
| **Eye gaze** | small | 1.21 (0.48) | 1.56 (0.6) | 2.76 (1.11) | 46.43 (34.68) | 1.52 (1.83) |
| | medium | 0.58 (0.47) | 1.62 (0.89) | 2.19 (1.03) | 19.59 (17.80) | 0.60 (1.04) |
| | large | 0.33 (0.15) | 1.71 (1.03) | 2.05 (1.04) | 11.15 (11.49) | 0.36 (0.80) |
| **Head gaze** | small | 1.15 (0.94) | 1.71 (0.54) | 2.86 (0.96) | 48.97 (33.39) | 1.30 (1.54) |
| | medium | 0.72 (0.50) | 1.69 (0.37) | 2.41 (0.6) | 36.34 (22.91) | 0.64 (0.84) |
| | large | 0.4 (0.30) | 1.75 (0.45) | 2.15 (0.45) | 26.42 (12.58) | 0.28 (0.53) |

## 5.2 Head Movement

Observing the head movement that participants made while performing an action on a key of the keyboard UI element (see Figure 6, left), the least head movement was required while using the eye-gaze method on the medium FOV with an average of 2.7 ° of head movement, closely followed by the eye-gaze method on the large FOV. Participants required the most head movement using the head-gaze method on the small FOV with an average of 17.5 °. On average, participants conducted 64.6% (10.6 °) less head movements using eye-gaze instead of head-gaze when interacting with the keyboard.

Observing the head movement on the menu UI element (see Figure 6, right), the least average head movement was required when using the eye-gaze method on the large FOV with an average of 11.1 ° of head movement. The most head movement was required while using the eye-gaze method on the small FOV with an average of 46.4 ° of head movement and the head-gaze method on the small FOV with an average of 49 °. When interacting with the menu, head movements were on average reduced by 31% (11.6 °) using eye-gaze.

The ANOVA revealed significant main effects of interaction method and FOV size in the keyboard scenario as well as in the menu scenario, as well as an interaction effect between them ($p < .001$). The post-hoc tests showed that in the keyboard condition, there was a significant difference between the small FOV and the large FOV ($p < .001$) and between the small FOV and medium FOV ($p = .008$) when using head-gaze. Using eye-gaze, all results differed significantly ($p < .001$) except the difference between medium FOV and large FOV. The time differences between eye gaze and head gaze interaction were significant over all FOV sizes ($p < .001$). In the menu scenario, the post-hoc test showed significant differences for all conditions ($p < .001$, $p = .014$ for eye gaze vs. head gaze in the small FOV).

Head- and eye movements were also analyzed with regard to the distance between the last UI element and the next desired element.

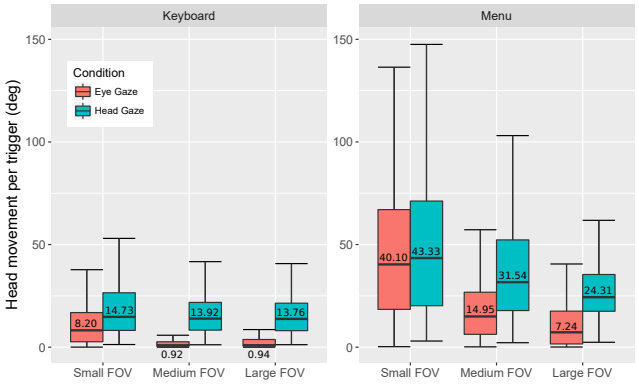


**Figure 6: The head movement participants conducted to trigger a UI element.**

Figure 7 shows locally weighted regression splines [4] for different relationships for interaction with the keyboard UI: The green spline shows the head movement which was conducted, the red line shows the eye movement and the blue line shows the combined angle of head movement and eye movement. For describing the relationships, we fitted robust linear models [36] which can handle the skewness of the data. For all conditions, these revealed that the distance of UI elements predict the amount of head movements and eye movements made ($p < .001$). Moreover, Figure 7 visualizes that in all eye-gaze conditions, there is less head movement than eye movements (even in the small FOV condition), which is the opposite case when head-gaze is used.

## 5.3 Errors

During the tasks, the number of started dwell timers was recorded. As ideally, only the dwell timer for the target UI element should be started, falsely started dwell timers are considered as errors - which

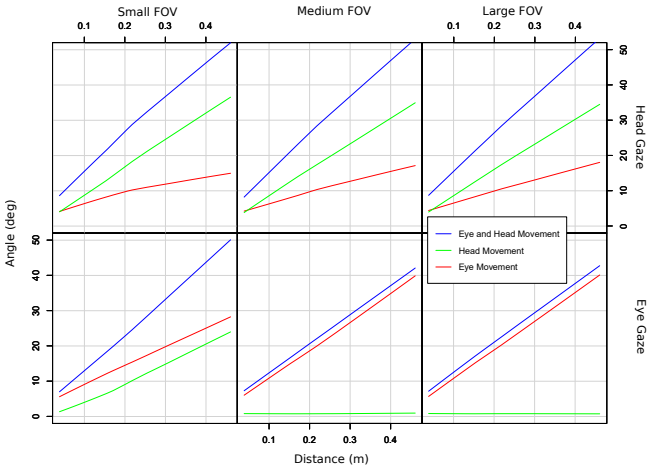


**Figure 7: The head movement and eye movement in relation for interaction with the keyboard UI.**

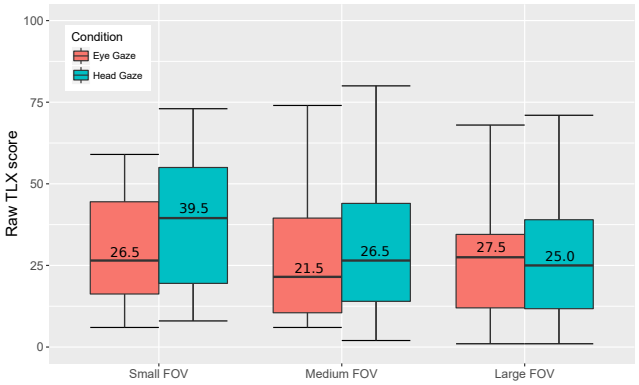


**Figure 8: The task load reported by the participants.**

however does not mean that the wrong UI element was triggered in the end.

Interacting with the keyboard UI, participants made least errors using eye-gaze with the medium and large FOV, for both on average 0.29 per selection. Using head-gaze, for medium FOV participants made 0.40 errors. With the small FOV, participant made on average 0.51 errors using eye-gaze and 0.56 using head-gaze. In case of the menu UI, participants made least errors using head-gaze in the large FOV condition, on average 0.28. Using eye-gaze, the average number of errors was 0.36 for this FOV size. Interacting with the small FOV, participants made most errors, on average 1.30 using head-gaze and 1.52 using eye-gaze.

The ANOVA revealed significant main effects of interaction method and FOV size in the keyboard scenario as well as in the menu scenario, as well as an interaction effect between them ($p < .001$). With the keyboard UI, for both eye-gaze and head-gaze there was no significant difference between large and medium FOV size. All other differences were significant ($p < .001$, $p = .005$ for eye gaze vs. head gaze in the small FOV). With the menu UI, head-gaze and eye-gaze differed significantly with the small FOV ($p = .019$), the medium FOV ($p = .057$) and the large FOV ($p = .057$). All other differences were highly significant ($p < .001$).

## 5.4 Task Load

Regarding the task load (see Figure 8), the results show that the eye-gaze method on the large FOV lead to the lowest average TLX score of 27 (SD=20.4) closely followed by the head gaze method on the large FOV with an average TLX score of 27.7 (SD=19.3). The eye gaze method on the medium FOV lead to an average TLX score of 29.8 (SD=21.5) and the eye gaze method on the small FOV to an average TLX score of 31.2 (SD=26.5). Participants furthermore reported an average TLX score of 32.8 (SD=26.5) for the head-gaze method on he medium FOV and the highest average TLX score of 36.8 (SD=39.5) for the head-gaze method on the small FOV size. An ANOVA did not show any significant differences between these task load values.

## 5.5 Qualitative Results

After participating in the experiment, 15 out of the 24 participants reported to prefer the eye-gaze interaction method. The most common reported reasons were that the eye-gaze interaction method was less exhausting to use (60%), easier to use (33.3%) or felt more natural (33.3%). While all of them reported to prefer this interaction method for all the FOV sizes, 7 participants stated that, with larger FOV sizes, the method would get even less exhausting as less head movement was required.

In contrast to that, 7 out of the 24 participants reported to prefer the head-gaze interaction method. Reasons for that were, amongst others: it felt more accurate (42.9%), was less exhausting (42.9%) and, that they felt that blinking would restart the dwell timer, making it more frustrating (42.9%). All 7 participants reported to prefer this method for all FOVs.

# 6 DISCUSSION

With this paper, we substantiate the question whether eye-gaze-based aiming is beneficial for interaction in AR or VR HMDs. In particular, we were interested in the effect of different FOVs.

Regarding performance, aiming using eye-gaze significantly outperformed head-gaze in terms of time-on-task and head movement. In the keyboard conditions, using eye gaze reduced time-on-task on average by 31.8% off the time required for aiming. In the menu scenario it had a benefit of 11.7% of the pure aiming time.

The full visibility of the the user interface makes a significant difference. In the keyboard scenario, the full keyboard was visible in both large and medium FOV conditions. Thus participants were able to instantly fixate the target UI elements with their eyes, which explains why there is no significant difference between these conditions. Because little to no head movement was required, the faster eye movement lead to a major speed advancement compared to head-gaze interaction.

We expected that this advantage will not play of in the small FOV condition, which reflects the current state of the art for AR devices. The partial visibility of the user interface due to the limited FOV made it necessary for users to conduct scanning movements with their heads to uncover the full UI and find the target. The results show that this condition was indeed more demanding, as head and

eyes had to be moved. The performances achieved with small FOV were significantly different from those with larger FOVs. However, even in the small FOV conditions, eye-gaze was significantly faster than head-gaze.

One concern regarding eye tracking was that it might be more demanding for the participants. In our set-up, however, we found no significant differences in the task load participants reported using the NASA TLX questionnaire. On the one hand, this shows that the evaluated task was easy to solve for them. On the other hand, the results reveal that there was no measurable overhead in task load when using eye tracking as input for this kind of UI interaction. The numbers of errors made support this finding: In most cases, eye gaze input lead to less errors than head gaze.

When analyzing the physical efforts that are required to use the interface, required head movements can be used as one indicator, as moving the head requires more energy than moving the eyes. Participants conducted close to zero head movements in the large and medium FOV condition of the keyboard scenario. This was different in the menu scenario: even with a large FOV, the distance to the target element could be so large that a short head movement was required. Still, in all conditions eye-gaze interaction could significantly reduce head movements. The fitted robust linear models for the keyboard condition reveal that head movement increases linearly with target distance in all head-gaze conditions. In the eye-gaze condition, it only increases in the small FOV condition. However, the share of head movement for triggering a UI element is still lower than the share of eye movements. Thus, participants only moved their head if the target element was not visible for the eyes. The main part of the aiming action was conducted using eye movements.

The qualitative feedback was in line with the quantitative results and our expectations. Most participants preferred the eye-gaze interaction method and those who did not gave feedback indicating that their preference was more due to problems with the accuracy (due to tracking problems) than eye-gaze interaction in general. One important observation was that participants fell into two groups. Most participants did not have any problems using the eye-gaze interaction methods. Other participants, however, had problems with the accuracy that were not resolvable with recalibrating the eye tracker or they reported that blinking would restart the dwell time for them. Specifically wearing contact lenses turned out be a problem with the eye-tracking device.

In the work of Qian et al. [30], the authors reported the opposite outcome of our results. As validated in section 4.1, the SMI HTC integration we used is advanced compared to the FOVE system used by Qian et al. both for eye tracking as well as head tracking. Qian et al.'s expectations that eye tracking would outperform head-gaze were even in line with our hypothesis. They specifically state problems with the tracking quality of the eye tracker, in particular while simultaneously moving the head, and in general observable jitter of the eye-gaze. Thus their results might have been an artifact of the employed technology.

## 7 CONCLUSIONS

Head-gaze and eye-gaze was evaluated as input for dwell-time-based UI interaction, in particular object selection, in VR and AR. Two different scenarios were tested: A keyboard similar to the one used with the Microsoft HoloLens and a menu typical for VR applications. To investigate differences resulting from FOV sizes, AR glasses were simulated in VR with small, medium and wide FOV.

The results show that eye-gaze outperforms head-gaze in terms of time-on-task, conducted head movements and errors made in all conditions. Using eye-gaze participants had a significant average speed advantage of above 100 ms over all conditions, which is a huge benefit in the realms of user interfaces. The advantage of eye-gaze over head-gaze for object selection is significant in all FOV conditions. The results thus demonstrate that eye-gaze is beneficial for both VR headsets with a large FOV (90 ° have been tested here) as well as for AR devices with a smaller FOV (36 ° tested). The full benefits of eye-gaze can be played out if the user interface is fully visible. However, even when scanning movements with the head are required, eye-gaze still was shown to be superior to head-gaze. At the same time, the task load was not affected by using the eyes as input for interaction in any condition.

While all participants successfully used the eye-gaze-based interaction, there are two main consequences we derive from the qualitative feedback. First, we suggest that head-gaze interaction, or, if manual interactions are possible, controller-based interaction should always be provided as fallback in case the achievable accuracy is low or the precision is below a relevant threshold. Second, the algorithms used for eye-gaze-based interaction could be further optimized. While a tuning of the eye-tracking algorithms is restricted when using off-the-shelf systems, we believe that personalizing the filters (e.g. our histogram-based filter) or, if possible, adapting the layout and in particular the sizes of the UI elements to the measured accuracy of the device further increases in robustness. We thus propose that the amount of interference the histogram provides and the dwell time itself should be part of the individual calibration process for eye-gaze interaction implementations.

In particular in AR, users are typically busy with their physical tasks, e.g. in the factory or during maintenance tasks in the field. In these situations, the hands are often engaged with the task and not available for operating a digital device without impact on the task performance. Based on the findings in eye-gaze-based interaction research, which we have substantiated here for specific criteria relevant for AR, we can only recommend that future AR devices include means for eye tracking.

## RESOURCES

A detailed description of the complete setup, hardware and a download link to the AR-Simulator that was used in the study is provided as a Unity-Package under the following URL:

http://mixedreality.eyemovementresearch.com/

## ACKNOWLEDGMENTS

This research was partly supported by the Cluster of Excellence Cognitive Interaction Technology 'CITEC' (EXC 277) at Bielefeld University, which is funded by the German Research Foundation (DFG), and partly by the BMBF project "ADAMAAS".

## REFERENCES


[1] Tobii AB. 2015. Tobii Pro VR Integration based on HTC Vive HMD. (Oct. 2015). https://www.tobiipro.com/product-listing/vr-integration/

[2] Kevin Wayne Arthur. 2000. *Effects of field of view on performance with head-mounted displays.* Ph.D. Dissertation. University of North Carolina at Chapel Hill. http://wwwx.cs.unc.edu/Research/eve/dissertations/2000-Arthur.pdf

[3] ASTRI. 2018. OmniVision and ASTRI Create Production-Ready Reference Design for AR Glasses. (2018). https://www.astri.org/news-detail/omnivision-and-astri-create-production-ready-reference-design-for-ar-glasses/

[4] William S Cleveland. 1981. LOWESS: A program for smoothing scatterplots by robust locally weighted regression. *The American Statistician* 35, 1 (1981), 54–54.

[5] Meta Company. 2017. Meta | Augmented Reality. (2017). http://www.metavision.com

[6] Nathan Cournia, John D. Smith, and Andrew T. Duchowski. 2003. Gaze- vs. Hand-based Pointing in Virtual Environments. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems (CHI EA '03)*. ACM, New York, NY, USA, 772–773. https://doi.org/10.1145/765891.765982

[7] Paul M. Fitts. 1954. The Information Capacity of the Human Motor System in Controlling the Amplitude of Movement. *Journal of Experimental Psychology* 47, 6 (1954), 381.

[8] Brian Guenter, Mark Finch, Steven Drucker, Desney Tan, and John Snyder. 2012. Foveated 3D Graphics. *ACM Trans. Graph.* 31, 6 (Nov. 2012), 164:1–164:10. https://doi.org/10.1145/2366145.2366183

[9] John Paulin Hansen, Kristian Tørning, Anders Sewerin Johansen, Kenji Itoh, and Hirotaka Aoki. 2004. Gaze Typing Compared with Input by Head and Hand. In *Proceedings of the 2004 Symposium on Eye Tracking Research & Applications (ETRA '04)*. ACM, New York, NY, USA, 131–138. https://doi.org/10.1145/968363.968389

[10] Tobias Höllerer and Steve Feiner. 2004. Mobile augmented reality. *Telegeoinformatics: Location-Based Computing and Services. Taylor and Francis Books Ltd., London, UK* 21 (2004). 00533.

[11] Felix Hülsmann, Timo Dankert, and Thies Pfeiffer. 2011. Comparing gaze-based and manual interaction in a fast-paced gaming task in Virtual Reality. In *Proceedings of the Workshop Virtuelle & Erweiterte Realität 2011.* https://pub.uni-bielefeld.de/publication/2308550

[12] FOVE Inc. 2017. FOVE. (2017). https://www.getfove.com/

[13] Sensomotoric Instruments. 2016. Imprint SensoMotoric Instruments. (2016). https://www.smivision.com/eye-tracking/imprint

[14] Robert J. K. Jacob. 1990. What You Look at is What You Get: Eye Movement-based Interaction Techniques. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '90)*. ACM, New York, NY, USA, 11–18. https://doi.org/10.1145/97243.97246

[15] C. Lee, S. Bonebrake, D. A. Bowman, and T. Höllerer. 2010. The role of latency in the validity of AR simulation. In *2010 IEEE Virtual Reality Conference (VR)*. 11–18. https://doi.org/10.1109/VR.2010.5444820

[16] C. Lee, S. Bonebrake, T. Hollerer, and D. A. Bowman. 2009. A Replication Study Testing the Validity of AR Simulation in VR for Controlled Experiments. In *2009 8th IEEE International Symposium on Mixed and Augmented Reality*. 203–204. https://doi.org/10.1109/ISMAR.2009.5336464

[17] C. Lee, G. A. Rincon, G. Meyer, T. Höllerer, and D. A. Bowman. 2013. The Effects of Visual Realism on Search Tasks in Mixed Reality Simulation. *IEEE Transactions on Visualization and Computer Graphics* 19, 4 (April 2013), 547–556. https://doi.org/10.1109/TVCG.2013.41

[18] Jae-Young Lee, Hyung-Min Park, Seok-Han Lee, Soon-Ho Shin, Tae-Eun Kim, and Jong-Soo Choi. 2011. Design and implementation of an augmented reality system using gaze interaction. *Multimedia Tools and Applications* 68, 2 (Dec. 2011), 265–280. https://doi.org/10.1007/s11042-011-0944-5

[19] Marc Levoy and Ross Whitaker. 1990. Gaze-directed Volume Rendering. In *Proceedings of the 1990 Symposium on Interactive 3D Graphics (I3D '90)*. ACM, New York, NY, USA, 217–223. https://doi.org/10.1145/91385.91449 00137.

[20] LooxidLabs. 2017. LooxidLabs HomePage. (2017). http://looxidlabs.com/

[21] I. Scott MacKenzie. 1992. Fitts' Law As a Research and Design Tool in Human-computer Interaction. *Hum.-Comput. Interact.* 7, 1 (March 1992), 91–139. https://doi.org/10.1207/s15327051hci0701_3

[22] Päivi Majaranta and Richard Bates. 2009. Special issue: Communication by gaze interaction. *Universal Access in the Information Society* 8, 4 (March 2009), 239–240. https://doi.org/10.1007/s10209-009-0150-7 00000.

[23] Päivi Majaranta and Andreas Bulling. 2014. Eye Tracking and Eye-Based Human–Computer Interaction. In *Advances in Physiological Computing*. Springer, London, 39–65. https://link.springer.com/chapter/10.1007/978-1-4471-6392-3_3

[24] Martin Meißner, Jella Pfeiffer, Thies Pfeiffer, and Harmen Oppewal. 2017. Combining virtual reality and mobile eye tracking to provide a naturalistic experimental environment for shopper research. *Journal of Business Research* (2017). https://doi.org/10.1016/j.jbusres.2017.09.028

[25] Cem Memili, Thies Pfeiffer, and Patrick Renner. 2016. Bestimmung von Nutzerpräferenzen für Augmented-Reality Systeme durch Prototyping in einer Virtual-Reality Simulation. In *Virtuelle und Erweiterte Realität - 13. Workshop der GI-Fachgruppe VR/AR*. https://pub.uni-bielefeld.de/publication/2905704

[26] Microsoft. 2016. Microsoft HoloLens. (2016). https://www.microsoft.com/de-de/hololens

[27] Darius Miniotas. 2000. Application of Fitts' Law to Eye Gaze Interaction. In *CHI '00 Extended Abstracts on Human Factors in Computing Systems (CHI EA '00)*. ACM, New York, NY, USA, 339–340. https://doi.org/10.1145/633292.633496

[28] Susanna Nilsson, Torbjörn Gustafsson, and Per Carleberg. 2009. Hands Free Interaction with Virtual Information in a Real Environment: Eye Gaze as an Interaction Tool in an Augmented Reality System. *PsychNology Journal* 7, 2 (Aug. 2009), 175–196.

[29] Hyung Min Park, Seok Han Lee, and Jong Soo Choi. 2008. Wearable Augmented Reality System Using Gaze Interaction. In *Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR '08)*. IEEE Computer Society, Washington, DC, USA, 175–176. https://doi.org/10.1109/ISMAR.2008.4637353

[30] Yuan Yuan Qian and Robert J. Teather. 2017. The Eyes Don'T Have It: An Empirical Comparison of Head-based and Eye-based Selection in Virtual Reality. In *Proceedings of the 5th Symposium on Spatial User Interaction (SUI '17)*. ACM, New York, NY, USA, 91–98. https://doi.org/10.1145/3131277.3132182

[31] E. Ragan, C. Wilkes, D. A. Bowman, and T. Hollerer. 2009. Simulation of Augmented Reality Systems in Purely Virtual Environments. In *2009 IEEE Virtual Reality Conference*. 287–288. https://doi.org/10.1109/VR.2009.4811058

[32] Linda E. Sibert and Robert J. K. Jacob. 2000. Evaluation of Eye Gaze Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '00)*. ACM, New York, NY, USA, 281–288. https://doi.org/10.1145/332040.332445

[33] Erik Steindecker, Ralph Stelzer, and Bernhard Saske. 2014. Requirements for Virtualization of AR Displays within VR Environments. In *Virtual, Augmented and Mixed Reality. Designing and Developing Virtual and Augmented Environments*, Randall Shumaker and Stephanie Lackey (Eds.). Number 8525 in Lecture Notes in Computer Science. Springer International Publishing, 105–116. http://link.springer.com/chapter/10.1007/978-3-319-07458-0_11

[34] Vildan Tanriverdi and Robert J. K. Jacob. 2000. Interacting with Eye Movements in Virtual Environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '00)*. ACM, New York, NY, USA, 265–272. https://doi.org/10.1145/332040.332443

[35] Takumi Toyama, Daniel Sonntag, Jason Orlosky, and Kiyoshi Kiyokawa. 2015. Attention Engagement and Cognitive State Analysis for Augmented Reality Text Display Functions. In *Proceedings of the 20th International Conference on Intelligent User Interfaces (IUI '15)*. ACM, New York, NY, USA, 322–332. https://doi.org/10.1145/2678025.2701384

[36] William N Venables and Brian D Ripley. 2013. *Modern applied statistics with S-PLUS*. Springer Science & Business Media.

[37] A. M. Wafaa, N. D. Bonnefoy, E. Dubois, P. Torguet, and J. P. Jessel. 2008. Virtual Reality Simulation for Prototyping Augmented Reality. In *2008 International Symposium on Ubiquitous Virtual Reality*. 55–58. https://doi.org/10.1109/ISUVR.2008.9

[38] Colin Ware and Harutune H. Mikaelian. 1987. An Evaluation of an Eye Tracker As a Device for Computer Input. In *Proceedings of the SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface (CHI '87)*. ACM, New York, NY, USA, 183–188. https://doi.org/10.1145/29933.275627

[39] Jacob O Wobbrock, Leah Findlater, Darren Gergle, and James J Higgins. 2011. The aligned rank transform for nonparametric factorial analyses using only anova procedures. In *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, 143–146.