An Analysis of Eye-Tracking Data in Foveated Ray Tracing

Thorsten Roth[1,2], Martin Weier[1,3], André Hinkenjann[1], Yongmin Li[2], Philipp Slusallek[3,4,5]

Abstract— We present an analysis of eye tracking data produced during a quality-focused user study of our own foveated ray tracing method. Generally, foveated rendering serves the purpose of adapting actual rendering methods to a user's gaze. This leads to performance improvements which also allow for the use of methods like ray tracing, which would be computationally too expensive otherwise, in fields like virtual reality (VR), where high rendering performance is important to achieve immersion, or fields like scientific and information visualization, where large amounts of data may hinder real-time rendering capabilities. We provide an overview of our rendering system itself as well as information about the data we collected during the user study, based on fixation tasks to be fulfilled during flights through virtual scenes displayed on a head-mounted display (HMD). We analyze the tracking data regarding its precision and take a closer look at the accuracy achieved by participants when focusing the fixation targets. This information is then put into context with the quality ratings given by the users, leading to a surprising relation between fixation accuracy and quality ratings.

Index Terms—Ray tracing, eye-tracking, foveated rendering

1 Introduction and Related Work

One of the major goals of virtual reality is the ability to present a real or imagined world to the user in a compelling and convincing way. With head-mounted displays (HMDs) becoming widely available, a suitable display technology which enables an immersive presentation already exists. To guarantee a visually convincing experience, it is highly important to provide a sufficient display resolution. While early devices worked at low resolutions (Forte VFX 3D, 1997, $263 \times 480 \times 2 = 0.25$ million pixels), modern HMDs have improved much in this regard (StarVR, 2016, $2560 \times 1440 \times 2 = 7.37$ million pixels). Nevertheless, the resolution is still a limiting factor regarding immersion: According to Warren Hunt [5], it would be necessary to render at a resolution of $32\text{k} \times 24\text{k} = 768$ million pixels for the full dynamic field of view (200° horizontally, 150° vertically) to match the full retinal resolution. This is two orders of magnitude more than current HMDs.

As rendering at such resolutions interactively is far beyond reach of current and foreseeable hardware and software solutions, it is necessary to develop methods which enable us to use increasing display resolutions while maintaining update rates. This challenge can be approached by adopting techniques from the field of foveated rendering, where a user's gaze and the perceptual limitations of human vision are exploited in order to adaptively adjust rendering quality across the image plane. According to [14], central vision in humans happens within the areas of the fovea (up to 5.2° from the optical axis), the parafovea (up to 9°) and the perifovea (up to 17°), while vision in areas further away from the central optical axis is referred to as peripheral vision.

While early foveated rendering methods did not employ eye-tracking and made use of focus assumptions [3] and models of visual attention [1, 17], such systems have the inherent limitation of being unable to predict the user's focus exactly. Most foveated rendering methods so far employed rasterization (e.g., [4]), but some ray-based methods have also been introduced. Fujita et al. [2] use a precomputed sampling pattern and kNN-based methods to reconstruct an image from sparse samples. Pohl et al. [8] present a system for exploiting lens astigmatism in HMDs. Their rendering system consists of a CPU-based ray tracer and ray casting, but it does not adapt to the content or the user's gaze. Stengel et al. [10] present a system that reduces the amount of shading in rasterization by accounting for visual acuity, eye motion, contrast and brightness adaptation. Our system is focused mainly on the reduction of the amount of traced rays, which inherently includes shading.

The data presented in this paper is based on the foveated ray tracing technique presented in [16], which allows for a fully adaptive adjustment of rendering quality across the image plane without the necessity of rendering large areas at uniform resolutions. In addition, we incorporate a reprojection technique to improve image quality and fill gaps in the sparsely sampled image. This is related to methods like Walter et al.'s Render Cache [13], the Holodeck Ray Cache [15], the Tapestry system [9], the Shading Cache [12] and the system by Jeschke et al. presented in [6]. However, what makes our system unique is the combination of a performance-focused reprojection method based on a coarse geometry approximation with foveated rendering methods to achieve visually pleasant results that can be rendered very fast.

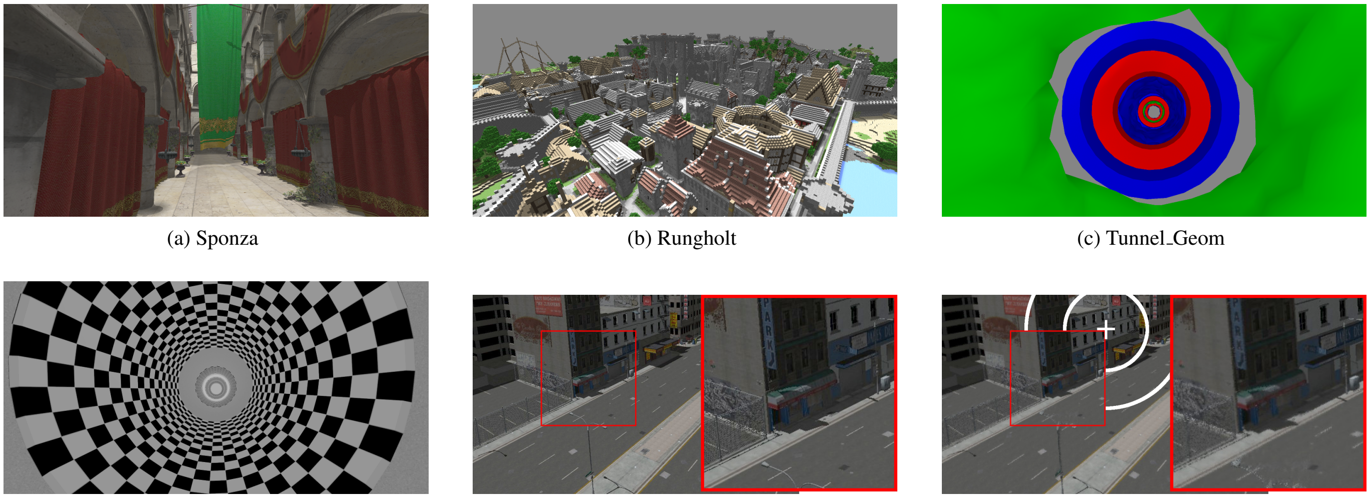
Our rendering algorithm is fully parameterized by the user's gaze, which is determined using a binocular SMI eye-tracker built into an Oculus Rift DK2. We conducted user studies that have shown the visual quality achieved by our system to be mostly indiscernible from full ray tracing while its rendering performance is clearly superior.

While accounting for the perceived rendering quality is extremely important for developing perception-aware rendering methods, this paper serves the purpose of taking a closer look at the recorded eye-tracking data of participants and their relation to the outcome of the study. In order to do this, it is necessary to understand the basic design of the user study and to know which data has been recorded, which is described in Section 2. Our approach to analyzing this data as well as the results of this analysis are presented in Section 3, revealing a surprising relation between fixation accuracy and quality ratings.

The key contribution of our work is to demonstrate how the presence of effects like visual tunneling can influence a user's perception in a foveated rendering setup in a way that allows for altering rendering quality to improve rendering performance in certain scenarios. This is done by looking into the user's ability to focus moving and static fixation targets in various scenes. The focusing accuracy is then determined by comparing the fixation targets with the recorded tracking data. Finally, comparing this accuracy with quality ratings given by participants provides possible evidence of the presence of visual tunneling effects [7] and the magnitude of their influence on the users' perception. Another contribution is an estimation of eye-tracking precision towards outer image areas. This information is also used to filter the tracking data. We draw conclusions and make suggestions how to benefit from our findings in practical systems in Section 4.

- *Thorsten Roth is with Bonn-Rhein-Sieg University of Applied Sciences and Brunel University London. E-mail: thorsten.roth@h-brs.de.*
- *Martin Weier is with Bonn-Rhein-Sieg University of Applied Sciences and Saarland University. E-mail: martin.weier@h-brs.de.*
- *André Hinkenjann is with Bonn-Rhein-Sieg University of Applied Sciences. E-mail: andre.hinkenjann@h-brs.de.*
- *Yongmin Li is with Brunel University London. E-mail: yongmin.li@brunel.ac.uk.*
- *Philipp Slusallek is with Saarland University, Intel Visual Computing Institute and German Research Center for Artificial Intelligence (DFKI). E-mail: philipp.slusallek@dfki.de.*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx



(a) Sponza

(b) Rungholt

(c) Tunnel_Geom

(d) Tunnel_Maps

(e) Full Ray Tracing

(f) Foveated rendering

Fig. 1: (a) to (d): Scenes used for user studies of our implementations. (e), (f): A direct comparison between full ray tracing and foveated rendering including the foveal region configuration.

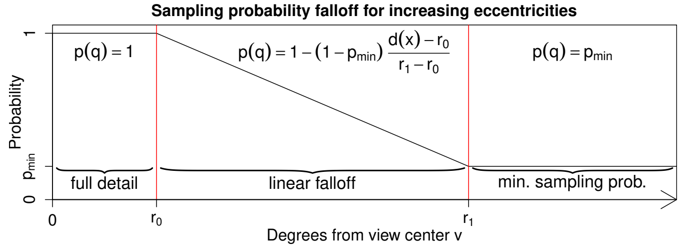


Fig. 2: Foveal function determining the sampling probability per pixel as a function of eccentricity. Parameters (r_0, r_1, p_{min}) are freely adjustable [16].

2 Rendering Process and Test Setup

Ray-based methods allow for fully adaptive sampling of the image plane. Therefore, we dynamically adjust the sampling probability of each individual pixel by accounting for its *eccentricity* (angular distance to the user's gaze). This adjustment is based on a freely parameterizable falloff function (the *foveal function*). While the physiology of the human eye shows a hyperbolic falloff in visual acuity with increasing distances to the fovea, this only matches the receptor density of cones, which are responsible for color perception. The density of rods, which enable brightness perception, decreases in a much more linear fashion [11]. Also, a linear model matches visual acuity well for small angles [4]. As this results in humans being very sensitive to flickering and motion in the peripheral areas of the visual field, we decided to adopt a piecewise linear foveal function instead of a hyperbolic falloff. This function is shown in Fig. 2.

The foveal function is used to make a sampling decision for each individual pixel in an image for each rendering iteration. This yields a decreased coverage of the image with pixel information towards the outer regions, making it necessary to fill in the gaps. To do this, we employ a reprojection method. First, color and geometry information are always computed for a uniformly sampled low-resolution version of the image (referred to as *support image* and *support G-buffer*). This information is used to generate a coarse, view-dependent mesh which is overlayed with the previous frame's color information and then reprojected to the new view. Reprojection errors due to disocclusions or movement and areas with insufficient quality are filled with information from the support image, while areas around geometric discontinuities are fully resampled.

Depending on the chosen parameters for the foveal function (referred to as the *foveal region configuration* or *FRC*), speedups compared to full ray tracing may vary. For a medium-sized FRC, which has exhibited good perceptual results in our user study, a speedup of 1.46 to 1.92 is achieved for scenes illuminated with a single point light. For one area light with 8 samples per pixel (spp), these increase to a range of 2.52 to 3.52, while using ambient occlusion with 16 spp yields speedups between 3.02 and 4.18 depending on the scene. Speedups have been determined for a single standard flight through the scenes *Sibenik*, *Sponza*, *Rungholt* and *Urban Sprawl*. Measurements have been performed on a standard PC with an Intel Core i7-3820 CPU, 64GiB of RAM and an NVIDIA GeForce Titan X graphics card at a resolution of 1182×1464.

To verify the visual quality of our method, we conducted a user study which was designed as a within-subject study with a $4 \times 4 \times 3$ full factorial design. Each participant completed 96 trials in randomized order, consisting of a full factorial combination of four scenes {*Sponza*, *Tunnel_geom*, *Tunnel_maps*, *Rungholt*} (see Fig. 1), four FRCs {small, medium, large, full} (described below) and three fixation types {fixed, moving, free}. All conditions were presented twice. Full ray tracing was included as the FRC *full*, representing our control group. Each trial consisted of an 8-second-flight with one specific factor combination. The utilized FRCs consisted of parameter triplets for the foveal function: *small* $(r_0 = 5°, r_1 = 10°, p_{min} = 0.01)$, *medium* $(10°, 20°, 0.05)$, *large* $(15°, 30°, 0.1)$ and *full* $(\infty, \infty, 1)$.

Fixation types were varied in order to determine the influence of visual tunneling effects. The *fixed focus* mode included a fixation cross at the image center which users had to focus the entire time, while the *moving target* mode included the task of focusing a green, moving sphere. This sphere's motion was tied to randomly generated paths across the image area and happened at static velocities. We varied the utilized paths in all trials except in repetitions to avoid learning effects. However, they were identical for all subjects. The foveal region was always centered around the fixation target for both modes. The *free focus* mode allowed users to freely look around in the image. It is left out in the remainder of this paper as it does not provide any reference for comparing recorded gaze coordinates. Nevertheless, the quality ratings for this data could be analyzed based on our findings regarding tracking precision.

Users had to rate the perceived quality as their agreement to the statements "The shown sequence was free of visual artifacts." and "I was confident of my answer." on a 7-point Likert scale from *strongly disagree* (-3) to *strongly agree* (3). During all trials, we recorded the tracking data to determine whether users followed the prescribed paths. Rendering was performed at the Oculus Rift DK2's full update rate of 75Hz, while eye-tracking had an asynchronous update rate of 60Hz.

Performing an ANOVA and subsequent t-tests on the user study data has shown that users could not reliably differentiate between full and foveated ray tracing with a FRC of *medium* or *large* (the quality ratings given by the users showed no statistically significant differences for these FRCs). The significant main effect of fixation types suggested the presence of visual tunneling effects that made some amount of visual artifacts imperceivable to the users. Over all scenes, the moving target mode was rated significantly better ($M = 0.99$, $SD = 1.63$) than the static ($M = 0.43$, $SD = 1.81$) and free ($M = 0.43$, $SD = 1.89$) fixation modes. A detailed description of both the rendering algorithm and the user study as well as its results can be found in [16].

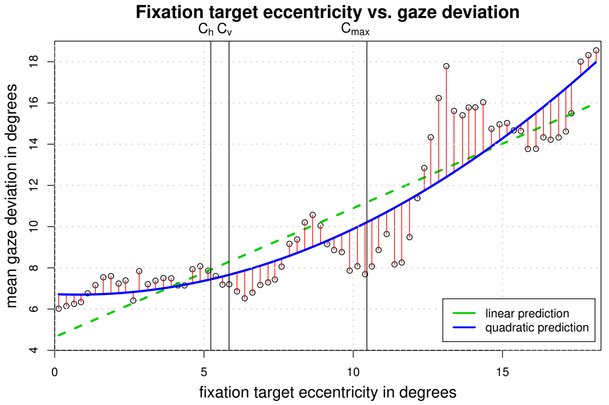


Fig. 3: Relation between tracking precision and fixation target's distance to the image center. C_h, C_v and C_{max} denote the extents of the area used by the eye tracker's calibration method horizontally, vertically and diagonally. The dotted green line represents the result of linear regression with a static slope, while the blue line represents linear regression with a quadratic equation. Residuals are shown in red.

3 Analysis and Results

Our goal is to analyze the tracking data and the user's corresponding quality ratings for the presence of effects like visual tunneling. Such effects may influence quality ratings in a way that yields results which would otherwise be unexpected when taking a look at the raw data.

3.1 Tracking precision

Before analyzing the recorded data any further, we need to ensure its validity. Eye-tracking precision suffers significantly towards outer image areas, which is also why calibration of the eye tracker only takes a relatively small area around the image center into account. In order to estimate the tracking precision, we looked at the distance between the recorded gaze and the fixation target's position in the image for each frame. We assume that the basic statistical distribution of the user's gaze relative to the fixation target is largely independent of the fixation target's position in the image. The analysis of the user study in [16] is based on the same assumption.

Let $F_{p,t}(i)$ be the fixation target's distance to the image center and $G_{p,t}(i)$ the recorded distance between the gaze and the fixation target for participant p in trial t at frame i. First, the data is sorted and averaged into bins of width $w = 0.25°$ depending on the fixation target position, so that we get $n = \lceil \max(F_{p,t}(i))/w \rceil$ bins $B_j = (\bar{F_j}, \bar{G_j}), 0 \le j < n$ with

$$F_j = \{F_{p,t}(i) \,|\, j \cdot w \le F_{p,t}(i) < (j+1) \cdot w\}, \tag{1}$$

$$G_j = \{G_{p,t}(i) \,|\, F_{p,t}(i) \in F_j\} \tag{2}$$

and $\bar{F_j}$ and $\bar{G_j}$ as the average values for the according bin. G_j can be interpreted as an approximate inverse measure of the tracking quality for eccentricity $j \cdot w$. We are now interested in the relation between G_j and F_j. To achieve more information about this relation, we perform a linear regression analysis. We assume that the relation between the fixation target's position and the tracking precision can be described by $\hat{G_j} = \beta_0 + \beta_1 F_j + \beta_2 F_j^2$. Linear regression on the prepared data yields $\beta = (6.723, -0.058, 0.037)$ with an R^2 value of 0.8317 and a correlation of 0.912. The p-values for the constant and square terms show their statistical significance ($p \approx 0$), while the linear term is not significant with $p = 0.679$. Fig. 3 shows the linear and the quadratic predictions for the gaze deviation. From the regression result the decrease in tracking precision with increasing eccentricities of the fixation target is clearly visible. Also, the figure includes the maximum extents of the area used by the eye tracker's calibration method. It is visible that the precision drops rapidly only a few degrees away from this area. Note that the results cannot be directly interpreted as the eye tracker's exact precision, as they include saccadic eye movements and latency-based deviations. However, they enable us to estimate the distance-dependent falloff in tracking precision.

Due to the decreasing tracking quality in outer areas, we filter the data to only include fixation target positions within the area used for calibration before further analysis.

3.2 Fixation Accuracy

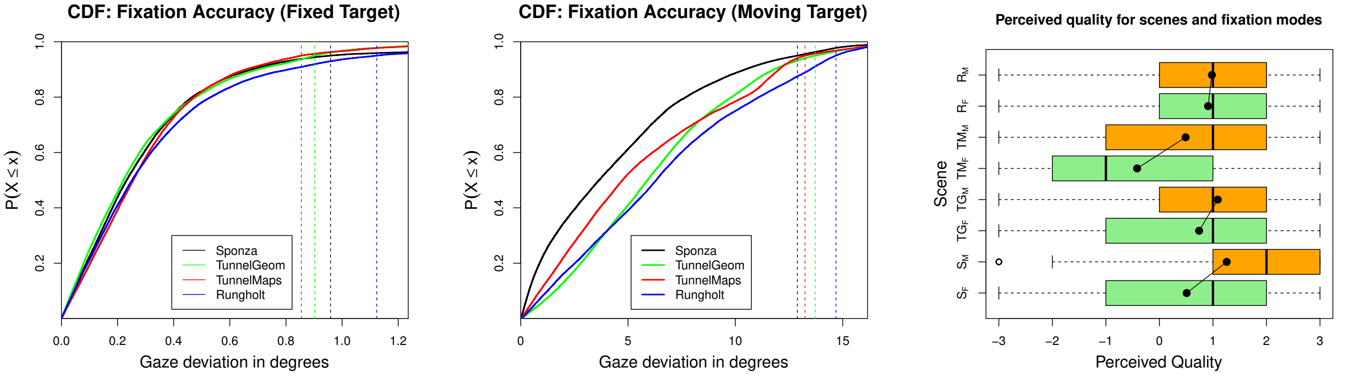
After filtering the recorded tracking data to only include the area of the image with good tracking reliability, we compare the participants' average fixation accuracy for the individual scenes. Figures 4a and 4b show the cumulative distribution functions (CDFs) for the fixed and the moving fixation target, respectively. The horizontal axis shows the angular distance between the user's gaze and the fixation target. While the 95% quantile of fixation accuracy was below 1.1° even in the scene with the worst fixation accuracy (Rungholt), the greatest 95% quantile was almost 15° for the moving target. 95% quantiles for the fixed target were all in [0.76, 1.03], while the moving target yielded 95% quantiles within [11.83, 14.59]. The much lower accuracy of the moving target fixation also means that the participants' gaze was often centered inside the border region or even outside of the area rendered all full detail. Consequently, parts of the image rendered at lower resolutions which have been reconstructed by our reprojection method were present in the area of central vision. Effects that may have caused the low accuracy include tracking latency and unpredictability of the moving target's path across the image plane.

3.3 Perceived Quality

Based on the data presented above, when it comes to rating the perceived quality for the fixed and moving targets, one would expect an inferior outcome for the latter. However, Fig. 4c unmistakably reveals that the opposite is the case: On average, the quality ratings for moving target fixation were better for all scenes. We interpret this as strong evidence for the presence of a visual tunneling effect, which means that human perception effectively filters the artifacts that are present in our foveated rendering system, making them largely imperceivable. Fig. 5 shows the distribution of gaze deviation, based on the data's 75% quartile. The color range indicates how often the users' gaze has been measured to be at the respective relative position to the fixation target (shown as *count* in the image). The illustrated deviation matches the data shown in Fig. 4. It becomes visible that there is a shift to the right for the deviation from the moving target. This is the case because our fixation paths were not equally distributed regarding the movement of the fixation target, which has moved left more often than right, causing the user's focus to be slightly shifted towards the right of the target on average.

4 Conclusion

We have presented an analysis of the tracking data recorded during our user study on foveated ray tracing. In addition to providing contextual information by giving an overview of our rendering method, we have shown that there is a significant decrease in tracking quality towards outer areas of the user's field of view. This is important for the implementation of foveated rendering algorithms as they need to be aware



(a)

(b)

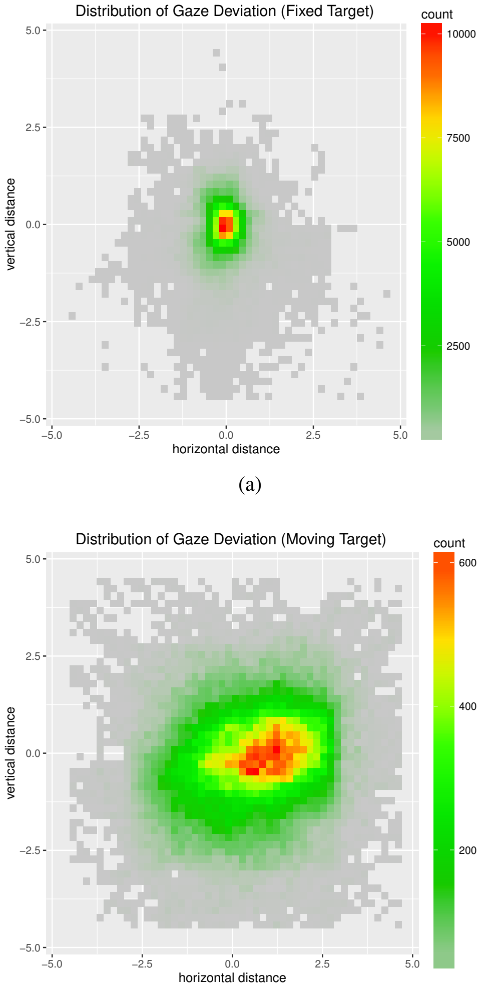
(c)

Fig. 4: (a) and (b) show the cumulative distribution functions (CDFs) for fixation accuracies measured for fixed and moving targets. Dotted lines show the 95% quantiles of gaze deviations for each scene. It is clearly visible that the fixation for the moving target has been much more difficult with 95% quantiles of deviations between 13 and 15 degrees. (c) shows the quality ratings given by the participants for the individual scenes and fixation modes. Although fixation accuracy has been much worse for the moving target, mean quality ratings were higher for all scenes. This is illustrated by the line segments between the two boxes for each scene, whose ends represent the respective mean values. S = Sponza, TG = TunnelGeom, TM = TunnelMaps, R = Rungholt. The subscript denotes whether the plotted data belongs to the moving (M) or the fixed target (F).

of the intersection between the image plane and the area of central vision in order to provide sufficient visual quality and avoid artifacts. We have found significant differences in the user's ability of focusing moving and static fixation targets. Though, while the user's tracked gaze was scattered over a far larger area around the fixation target for the moving fixation mode, the quality ratings given by the users seemed counterintuitive. For all scenes used during the study, the mean quality ratings were better for the moving fixation target, although the user's gaze did not match the fixation target, and thus the area rendered at full detail, very well. We attribute this to the presence of a strong visual tunneling effect induced by the task of following a moving target, effectively reducing the participant's field of view. The information presented in this paper can be utilized for further adaptation of foveated rendering methods, exploiting information about the potential presence of visual tunneling to improve rendering performance by adapting the visual quality level. This effectively supports the development of fast, high-quality rendering algorithms needed for high-resolution rendering at high frame rates which are necessary for providing a pleasant user experience when using HMDs for visualization in VR and other disciplines.

Acknowledgments

We would like to thank NVIDIA for providing us with two Quadro K6000 graphics cards for the user study, the Intel Visual Computing Institute, the European Union (EU) for the co-funding as part of the *Dreamspace* project, the German Federal Ministry for Economic Affairs and Energy (BMWi) for funding the *MATEDIS* ZIM project (grant no KF2644109) and the Federal Ministry of Education and Research (BMBF) for funding the project *OLIVE*.



(a)

(b)

Fig. 5: Distribution of gaze deviation for fixed (a) and moving (b) fixation targets.

References

[1] Eric Horvitz and Jed Lengyel. Perception, attention, and resources: A decision-theoretic approach to graphics rendering. In *UAI*, pp. 238–249. Morgan Kaufmann, 1997.

[2] M. Fujita and T. Harada. Foveated Real-Time Ray Tracing for Virtual Reality Headset, May 2014.

[3] T. A. Funkhouser and C. H. Séquin. Adaptive display algorithm for interactive frame rates during visualization of complex virtual environments. In *20th annual conference on Computer graphics and interactive techniques*, pp. 247–254. ACM, 1993.

[4] B. Guenter, M. Finch, S. Drucker, D. Tan, and J. Snyder. Foveated 3D Graphics. *ACM Transactions on Graphics*, 31(6):164, 2012.

[5] W. Hunt. Virtual Reality: The Next Great Graphics Revolution. Keynote Talk HPG, Aug 2015.

[6] S. Jeschke and M. Wimmer. Textured Depth Meshes for Real-time Rendering of Arbitrary Scenes. In *13th Eurographics Workshop on Rendering*, pp. 181–190. Eurographics Association, 2002.

[7] T. Miura. Coping with situational demands: A study of eye movements and peripheral vision performance. *Vision in Vehicles*, pp. 206–216, 1986.

[8] D. Pohl, T. Bolkart, S. Nickels, and O. Grau. Using astigmatism in wide angle HMDs to improve rendering. In *Virtual Reality*, pp. 263–264. IEEE, 2015.

[9] M. Simmons and C. H. Séquin. Tapestry: A Dynamic Mesh-based Display Representation for Interactive Rendering. In *Proceedings of the Eurographics Workshop on Rendering Techniques 2000*, pp. 329–340. Springer-Verlag, 2000.

[10] M. Stengel, S. Grogorick, M. Eisemann, and M. Magnor. Adaptive Image-Space Sampling for Gaze-Contingent Real-time Rendering. *Proc. Eurographics Conference on Rendering Techniques (EGSR) 2016*, 35(4), June 2016. Won the 'EGSR'16 Best Paper Award'.

[11] H. Strasburger, I. Rentschler, and M. Jüttner. Peripheral vision and pattern recognition: A review. *Journal of vision*, 11(5):1–82, jan 2011.

[12] P. Tole, F. Pellacini, B. Walter, and D. P. Greenberg. Interactive Global Illumination in Dynamic Scenes. *ACM Trans. Graph.*, 21(3):537–546, July 2002.

[13] B. Walter, G. Drettakis, and S. Parker. Interactive Rendering using the Render Cache. In D. Lischinski and G. Larson, eds., *Rendering Techniques (Proceedings of the Eurographics Workshop on Rendering)*, vol. 10, pp. 235–246. Springer-Verlag, Jun 1999.

[14] B. A. Wandell. *Foundations of Vision*. Stanford University, 1995.

[15] G. Ward and M. Simmons. The Holodeck Ray Cache: An Interactive Rendering System for Global Illumination in Nondiffuse Environments. *ACM Trans. Graph.*, 18(4):361–368, Oct. 1999.

[16] M. Weier, T. Roth, E. Kruijff, A. Pérard-Gayot, A. Hinkenjann, P. Slusallek, and Y. Li. Foveated real-time ray tracing for head-mounted displays. In *Accepted for Pacific Graphics 2016, Okinawa, Japan*, 2016.

[17] H. Yee, S. Pattanaik, and D. P. Greenberg. Spatiotemporal Sensitivity and Visual Attention for Efficient Rendering of Dynamic Environments. *ACM Trans. Graph.*, 20(1):39–65, Jan. 2001.