Using Astigmatism in Wide Angle HMDs to Improve Rendering

Daniel Pohl* — Intel Corporation
Timo Bolkart† — Saarland University
Stefan Nickels‡ — Intel Visual Computing Institute
Oliver Grau§ — Intel Corporation

Abstract

Lenses in modern consumer HMDs introduce distortions like astigmatism: only the center area of the displayed content can be perceived sharp while with increasing distance from the center the image gets out of focus. We show with three new approaches that this undesired side effect can be used in a positive way to save calculations in blurry areas. For example, using sampling maps to lower the detail in areas where the image is blurred through astigmatism, increases performance by a factor of 2 to 3. Further, we introduce a new calibration of user-specific viewing parameters that increase the performance by about 20-75%.

Index Terms: I.3.3 [Computer Graphics]: Picture / Image Generation—

1 Introduction

Combining smartphone panels with inexpensive optics (Figure 1) achieves immersivity with low cost HMDs [5, 6]. Unfortunately, these optics introduce distortions like radial and chromatic aberrations, which can be compensated in software [7], and astigmatism [3], which cannot be removed through software. Astigmatism (Figure 2) happens when the meridional and sagittal focal points do not coincide. It increases with the distance of the object point to the optical axis.

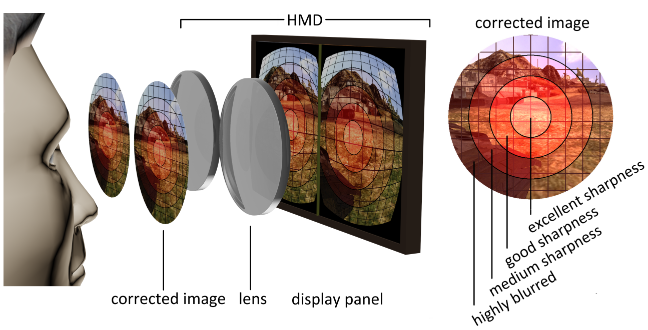


Figure 1: *Top: A low-cost immersive HMD. Image courtesy of [7]. Bottom: Effects of astigmatism schematically marked depending on sharpness.*

For high-quality rendering to a wide angle HMD software correction for radial and chromatic aberrations needs to be applied through barrel-distorting the image. Interactive applications rely on GPU-accelerated rasterization, which performs fast, but is limited to rendering a fixed rectangular grid. Warping the image after rendering results in undesired blur. Ray tracers or custom written rasterizers are usually slower, due to missing hardware acceleration. However, there it is possible to sample the grid barrel-distorted for rendering.

In this paper, we show that a calibration of the maximal visible area for users increases performance by 18-77%. Further, we show with three new methods that considering astigmatism of the lens can be used to increase performance. First, sampling maps to change the image quality depending on the amount of blur by using different numbers of rays per pixel. Second, we mix in a hybrid approach GPU-accelerated rasterization with ray tracing which leads to higher image quality at high frame rates. Third, we look at varying the quality of distortion shaders depending on the blur of the astigmatism.

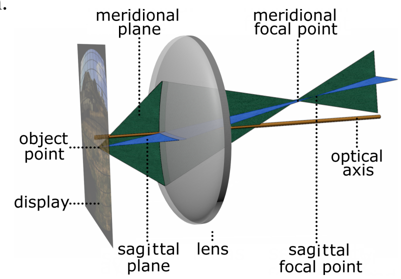


Figure 2: *Optical system with astigmatism.*

2 Related Work

Radial and chromatic aberrations in HMDs are handled in software [10, 5, 7]. In contrast to these methods, we also change the rendering for astigmatism. The idea of reducing rendering costs in areas where the human eye cannot perceive all details has been used by Tong and Fisher [8] for a flight simulator, which projected a large low-quality background image on a dome and added a smaller high-quality image where the user looks at. Levoy and Whitaker [4] change depending on the gaze the number of rays in a volume ray caster. Guenter et al. [2] follow the same idea depending on the user's gaze point. They use a rasterizer to generate three images at different sampling rates and composite them together. In contrast to our work, these approaches rely on eye tracking. Modern consumer HMDs do not support that. While future versions might add this, low cost devices like Google Cardboard [1] are unlikely to get that capability. Further, the effects of astigmatism have so far been ignored during rendering.

3 User-specific Calibration

We propose to measure the maximal radial area and its center that a user can see to optimize rendering. We visualize that area by a red circle with a blinking white border. The users were allowed to modify the center and the radius with the goal of not seeing the border and keeping the radius low. This is done first for each eye, then both. As the position of the HMD at the head might vary, we recommend to repeat the calibration after taking the HMD off and on. With two iterations this took about five minutes.

4 Rendering

Sampling Maps Our idea is based on Levoy and Whitaker [4], but works for HMDs without eye tracking. It uses the astigmatism of the lens for optimizations. Pixels further away from the center have higher blur, so we want to spend less rendering time there. We introduce the sampling map, a gray-scale texture at the size of the screen resolution in which a texel represents a scaling factor for the amount of samples per pixel to be used during rendering with supersampling. 0 for no samples, 255 as maximum. Our sampling map consists of ten parts at different radial distances to the lens center (Figure 3 left). We assigned every part a value, e.g. for the first two we assign the maximum and then decrease until 0. We linearly interpolate between the divided areas. Last, we tweak the values using high-frequency test data.

*e-mail: daniel.pohl@intel.com
†e-mail: tbolkart@mmci.uni-saarland.de
‡e-mail: nickels@intel-vci.uni-saarland.de
§e-mail: oliver.grau@intel.com

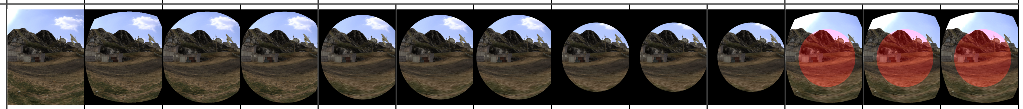
Area	Full	Oculus	Sampling Maps		User Lens "A"			User Lens "C"			Hybrid Rendering		
Rendering area													
Sampling Maps	✗	✗	✓	✓	✗	✓	✓	✗	✓	✓	✗	✓	✓
Adaptive Supersampling	✗	✗	✗	✓	✗	✗	✓	✗	✗	✓	✗	✗	✓
Max. Samples 1	54	60	—	—	71	—	—	97	—	—	116	—	—
Max. Samples 4	14	16	32	—	19	36	—	28	46	—	37	54	—
Max. Samples 16	3.9	4.4	13	21	5.2	13	22	7.6	15	27	11	18	35

Table 1: *Performance values in frames per second. Higher is better.*

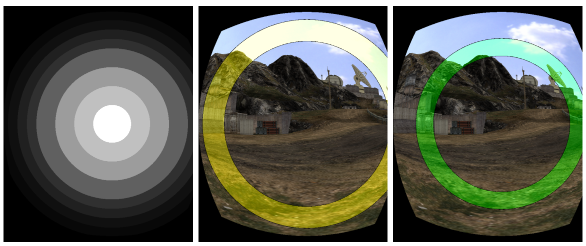


Figure 3: *Left: Sampling map. Center: Lowest and highest maximal radius of the visible area for lenses "A". Right: Analogue for lenses "C".*

We use two modes for supersampling. One has a fixed number of rays per pixel from the sampling map. The other uses adaptive supersampling [11]. Sampling maps are only applicable in renderers that allow choosing the amount of sampling on a per-pixel basis. It does not work with current GPU-accelerated rasterization.

Hybrid Rendering One can tradeoff between quality and performance for compensation of radial and chromatic distortions in HMDs by choosing between image warping on GPU-accelerated rasterizers or a custom renderer with a barrel-distorted sampling patterns to get better image quality, usually at lower performance [7]. Our new hybrid approach: first, the image is rendered using a GPU-accelerated rasterizer. Afterwards, we only render a radial region around the lens center with a ray tracer with the right sampling pattern to counter distortions. A GPU shader displays the rasterized image in the outer area, while warping it. The ray traced region is displayed directly in the inner area.

Distortion Shader Optimizations First, to pre-warp an image against optical distortions we use the maximal visible area that we got from the user-specific calibration and check in the distortion shader program if that area is relevant and otherwise stop processing. Second, as bicubic texture filtering during warping delivers superior quality compared to bilinear filtering [7], we use a hybrid shader: if the pixel is in the sharper area we use bicubic filtering, otherwise the faster bilinear texture lookups.

5 Experimental Results

System Description We render content from Enemy Territory: Quake Wars[1] at 1280×800 pixels with a ray tracer, accelerated by Intel Embree [9], to the Oculus Rift DK1 [5]. The PC has two Intel® Xeon® E5-2687W and a NVidia® GeForce® 680 GTX.

User-specific Calibration We use Rift lenses "A" (no vision correction) and "C" (highly shortsighted). Screen coordinates are from 0 to 1. The results for the maximal visible radius with 13 test subjects are an average of 0.49, STD 0.04, minimum at 0.44 and maximum at 0.55 for "A". Average for "C" is 0.40, STD 0.03, minimum 0.36, maximum 0.45. The values are visualized in Figure 3 center and right.

Performance The performance for sampling maps, adaptive supersampling and hybrid rendering are in Table 1. Results for distortion shaders are in Table 2.

[1] id Software and Splash Damage

Filtering	bilinear		bicubic		hybrid	
Spat. distortion	✓	✓	✓	✓	✓	✓
Chrom. aberr.	✗	✓	✗	✓	✗	✓
Area Oculus	2.04	2.10	2.45	3.04	2.29	2.55
Area Lens "A"	2.00	2.06	2.39	2.88	2.27	2.48
Area Lens "C"	1.93	1.96	2.17	2.44	2.10	2.28

Table 2: *Time in ms for distortion shaders. Lower is better.*

6 Conclusion

We have shown that by considering astigmatism in HMDs, it can be used to improve rendering. With sampling maps to change the quality depending on blur, we got speed-ups of 2 to 3. With the hybrid rendering approach, we achieved 2 to 3 times higher frames rates compared to a fully ray traced approach. Next, we showed optimizations for distortion shaders which saved 2-16% time. Our user-specific calibration to the maximal visible area gives speed-ups between 18-77%.

References

[1] Google. Google Cardboard, 2014. `https://cardboard.withgoogle.com/`.

[2] B. Guenter, M. Finch, S. Drucker, D. Tan, and J. Snyder. Foveated 3d graphics. ACM SIGGRAPH Asia, 2012.

[3] E. Hecht. *Optics*. Addison-Wesley, 2002.

[4] M. Levoy and R. Whitaker. Gaze-directed volume rendering. In *Proceedings of the 1990 Symposium on Interactive 3D Graphics*, I3D '90, pages 217–223. ACM, 1990.

[5] Oculus VR. Oculus Rift, 2014. `http://www.oculus.com/`.

[6] J. Olson, D. Krum, E. Suma, and M. Bolas. A design for a smartphone-based head mounted display. In *Virtual Reality Conference (VR), 2011 IEEE*, pages 233–234, 03 2011.

[7] D. Pohl, G. S. Johnson, and T. Bolkart. Improved pre-warping for wide angle, head mounted displays. In *Proceedings of the 19th ACM symposium on Virtual reality software and technology*, VRST '13. ACM, 2013.

[8] H. Tong and R. Fisher. *Progress Report on an Eye-Slaved Area-of-Interest Visual Display*. Defense Technical Information Center, 1984.

[9] I. Wald, S. Woop, C. Benthin, G. S. Johnson, and M. Ernst. Embree: A kernel framework for efficient cpu ray tracing. *ACM Trans. Graph.*, 33(4), July 2014.

[10] B. Watson and L. Hodges. Using texture maps to correct for optical distortion in head-mounted displays. In *Virtual Reality Annual International Symposium, 1995. Proceedings.*, pages 172–178, Mar 1995.

[11] T. Whitted. Improved illumination model for shaded display. *Communications of the ACM*, 23(6):343–349, 1980.