

# DEAMP: Dominant-Eye-Aware Foveated Rendering with Multi-Parameter Optimization

Zhimin Wang\*

Xiangyuan Gu†

Feng Lu‡

State Key Laboratory of VR Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing, China

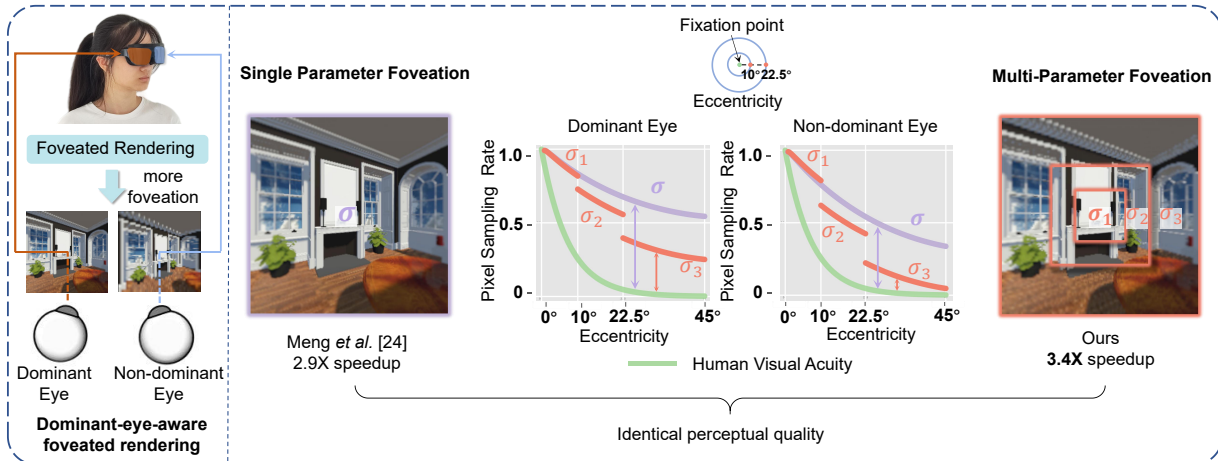


Figure 1: To accelerate the rendering of scenes, we propose a Dominant-Eye-Aware foveated rendering method optimized with Multi-Parameter foveation (DEAMP). Our method achieves average  $3.4\times$  speedup compared with the full-resolution rendering, while Meng *et al.* achieved  $2.9\times$  speedup [22], on binocular screens with a resolution of  $1600\times 1600$  per eye.

## ABSTRACT

The increasing use of high-resolution displays and the demand for interactive frame rates presents a major challenge to widespread adoption of virtual reality. Foveated rendering address this issue by lowering pixel sampling rate at the periphery of the display. However, existing techniques do not fully exploit the feature of human binocular vision, *i.e.*, the dominant eye. In this paper, we propose a Dominant-Eye-Aware foveated rendering method optimized with Multi-Parameter foveation (DEAMP). Specifically, we control the level of foveation for both eyes with two distinct sets of foveation parameters. To achieve this, each eye’s visual field is divided into three nested layers based on eccentricity. Multiple parameters govern the level of foveation of each layer, respectively. We conduct user studies to evaluate our method. Experimental results demonstrate that DEAMP is superior in terms of rendering time and reduces the disparity between pixel sampling rate and the visual acuity fall-off model while maintaining the perceptual quality.

**Index Terms:** Virtual Reality—Foveated Rendering—Eye tracking—Eye dominance

## 1 INTRODUCTION

Virtual reality (VR) generates immersive virtual environments by leveraging computer graphics. With the advent of the meta-verse in recent years, VR has established itself as a highly promising digital platform [6, 40, 43]. Nowadays, due to VR extensive applica-

tions across diverse fields, *e.g.*, entertainment, online education and healthcare, it has garnered significant attention [41, 42]. However, the heavy rendering overhead requires for high-quality VR content presents a major challenge to the widespread adoption of VR [39]. To address this issue, researchers are striving to enhance VR accessibility and affordability by exploring new methods and technologies to reduce rendering costs.

Foveated rendering aims to accelerate rendering while maintaining perceptual quality by leveraging the inherent feature of human visual system. Specifically, the fovea of human eyes, rich in optic nerves, offers detailed visual information, hence being pivotal for high-resolution visual tasks. On the contrary, the peripheral retina is less sensitive due to sparse distribution of optic nerves, thus incapable of delivering the same level of detail as the fovea [13, 25]. Therefore, employing high-quality rendering for the foveal region and low-quality rendering for the peripheral area conserves computational resources.

In recent years, numerous studies have investigated the foveated rendering techniques in VR [21, 35], primarily used in rasterization and ray tracing [39]. Using a deferred shading pipeline [33], rasterization transforms geometric information into geometric buffers (G-buffer), followed by the application of per-pixel lighting to the G-buffer [1]. To accelerate the rendering process, foveated rendering reduces the resolution of G-buffer and lowers the sampling rate at the periphery [10, 12, 23, 45]. Ray tracing rendering enables to control the number of rays emitted per pixel [11, 28, 38]. Higher numbers of rays result in better rendering quality, but also greater rendering costs. In this framework, foveated rendering can decrease the number of rays computed in the periphery of the visual field, thereby accelerating rendering [17, 37, 44].

Despite their exhaustive algorithmic optimizations, these foveated rendering techniques do not fully exploit the feature of human binocular vision, *i.e.*, the dominant eye [29] to accelerating the rendering. For instance, the dominant eye exhibits superior to the non-dominant

\*joint first author, e-mail: zm.wang@buaa.edu.cn

†joint first author, e-mail: guxy@buaa.edu.cn

‡corresponding author, e-mail: lufeng@buaa.edu.cn

eye in visual acuity [32], and color discrimination [18]. Previous studies have found that approximately 70% of the individuals are right-eye dominant, while 29% are left-eye dominant [7]. Taking advantage of the trait of binocular vision, rendering differing resolutions to each eye can significantly speed up processing compared to rendering identical resolution images in both eyes.

Meng *et al.* employed the eye-dominance-guided foveated rendering technique (EFR) [22]. They rendered the scene with higher level of foveation for the non-dominant eye than the dominant eye. Their findings showed improved acceleration compared to the original foveated rendering. Specifically, they applied single-parameter foveation across the entire visual field, as shown in Fig. 1. However, this method causes a significant gap between the pixel sampling rate and the visual acuity fall-off model at the periphery of the visual field. It results in an increased number of rendered pixels, leading to unnecessary rendering costs. Therefore, there is a need for better integration between foveated rendering and the theory of the dominant eye.

To address these issues, we propose a Dominant-Eye-Aware foveated rendering method optimized with Multi-Parameter foveation (DEAMP). Specifically, for the dominant eye and the non-dominant eye, we apply two distinct sets of foveation parameters, respectively, as shown in Fig. 1. For each individual eye, the entire visual field is divided into three eccentricity layers according to the eccentric angle from the fixation point. Each layer utilizes single-parameter foveation to control the level of foveation. Hence, multi-parameter foveation governs the level of foveation for the entire image of each eye. The optimization goal of multiple parameters is to narrow the gap between the pixel sampling rate and the visual acuity fall-off model in the identical perceptual quality. We conduct two user studies to evaluate our approach. The results show that our method significantly reduces the rendering time compared to existing methods. Our source code are publicly available at <https://github.com/CTofStars/DEAMP>.

In summary, the contributions of this paper are as follows:

- We propose a dominant-eye-aware foveated rendering method with multi-parameter foveation, *i.e.*, DEAMP, which significantly improves the rendering speed while guaranteeing visual quality.
- We conduct user studies to identify multiple levels of foveation for both the dominant and the non-dominant eye. Through comparison and analysis, we demonstrate the superiority of our method.
- Our method is implemented with the deferred shading pipeline on a GPU, and achieves  $3.40\times$  speedup compared with the full-resolution rendering at a resolution of  $1600 \times 1600$  per eye.

## 2 RELATED WORK

In this section, we review the recent studies in foveated rendering for VR and discuss the application of eye dominance.

### 2.1 Foveated Rendering

Foveated rendering techniques aim to reduce the computational budget required for high-quality VR content without affecting the perceptual quality evaluated by users [5, 26, 36], by taking cues from the human visual system. Firstly, we will introduce the human visual system and subsequently present the main categories of foveated rendering techniques.

*Human visual system.* Foveated rendering employs human visual features as its foundation [39]. Specifically, The fovea, located in the central region of the retina, has a dense concentration of optic nerves, allowing for highly detailed visual perception. In contrast, the peripheral region beyond the foveal area has fewer

optic nerves and is therefore less sensitive, resulting in lower visual acuity compared to the fovea [13, 25]. Visual acuity reduces by more than 95% from the central vision to the peripheral vision [39]. By using high-quality rendering in the foveal region and low-quality rendering in the peripheral area, this technique saves computing resources.

*Rasterization.* As reviewed by Wang *et al.* [39], approaches to foveated rendering mainly include rasterization, ray tracing, *etc.* In a deferred shading pipeline, the rasterization stores geometric information in geometric buffers (G-buffer), and calculate lights per pixels [1]. Foveated rendering reduces the resolution of G-buffer and lowers the sampling rate in the periphery to accelerate the rendering process. Guenter *et al.* divided the visual field around the gaze point into three eccentricity layers, and decreased the sampling rate of pixels as eccentricity increased [12]. This is comparable to our approach with respect to single-eye foveation but we fix the layer size and only adjust the sampling rate of each layer. Meng *et al.* introduced a two-pass pipeline for kernel foveated rendering (KFR). They used kernel log-polar transformation in the first pass to downsample the G-buffer resolution. The second pass upsampled the reduced-resolution rendering using inverse kernel log-polar transformation [23]. Ye *et al.* used a similar two-pass pipeline for foveated rendering but employed rectangular mapping image transformation to achieve a more refined foveation effect with minimal artifacts [45]. In addition to the spatial factor, some researchers have incorporated the temporal factor into foveated rendering [27, 34]. For instance, Franke *et al.* reused peripheral pixels by spatio-temporally reprojecting them from prior frames [10].

*Ray tracing rendering.* Ray tracing rendering allows for adjustment of the number of rays emitted per pixel [11, 28, 38]. While higher ray counts produce superior visual quality, they incur more considerable rendering expenses. In this paradigm, foveated rendering can reduce the number of rays calculated in the periphery of the visual field to accelerate rendering. Koskela *et al.* proposed a visual-polar space for foveated ray tracing and thus reduced the required number of rays [16].

*Other foveated rendering.* Apart from the aforementioned rendering paradigms, several studies have investigated other approaches for achieving foveated rendering, including neural rendering [9, 14] and photon mapping [31].

### 2.2 Eye Dominance

Recently, the theory of eye dominance is employed for further accelerating rendering. The eye dominance is the feature of human binocular vision. The dominant eye displays more sensitive than the non-dominant eye in scene perception [29]. Previous studies have found that approximately 70% of the individuals are right-eye dominant, while 29% are left-eye dominant [7]. Shneur *et al.* evaluated the role of eye dominance in non-rivalry conditions by comparing performance of the dominant or the non-dominant eye, when viewing the target with different size [32]. They found that the dominant eye have priority in visual processing and may inhibit information from the non-dominant eye. Koçtekin *et al.* found the dominant eye has priority in the *r/g* local color and concluded the dominant eye has the priority in color vision discrimination [18]. Oishi *et al.* found the dominant eye has the functional dominancy in horizontal saccades at reading distance [24].

Meng *et al.* employed the Eye dominance model on Foveated Rendering (EFR) technique [22]. They utilized the previously mentioned foveated rendering method KFR to create the scene for the dominant eye at a normal foveation level and generate the scene for the non-dominant eye with greater foveation level. The results showed the improved acceleration compared with the original foveated rendering. However, as Meng *et al.* applied single-parameter foveation across the entire visual field, as shown in Fig.1. This method creates a substantial disparity between the pixel sampling rate and the

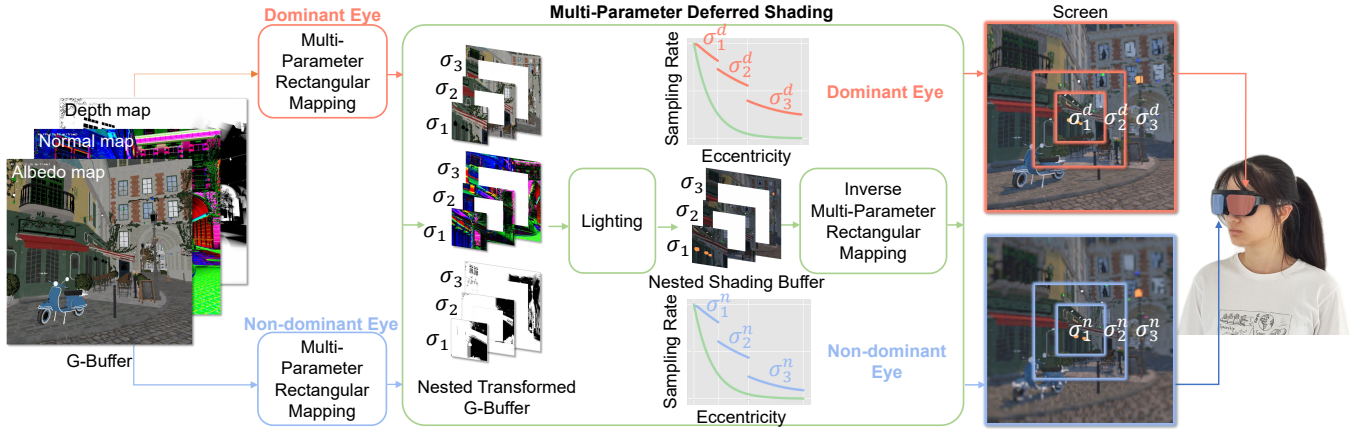


Figure 2: This pipeline of the DEAMP. Our method integrates with the deferred shading pipeline. We implement the Multi-Parameter (M-P) rectangular mapping for the G-buffer. During the lighting pass, we shade each pixel by utilizing the information from the transformed G-buffer. Finally, we apply the inverse M-P rectangular mapping of the nested shading buffer to the full-resolution screen.

visual acuity fall-off model in the visual field’s periphery. Consequently, this leads to a higher number of rendered pixels, resulting in extra rendering expenses. It’s worth mentioning that the KFR can approach to the visual acuity fall-off model by decreasing the parameter  $\alpha$  to 1. However, the KFR’s experiments have shown that decreasing  $\alpha$  results in the worse visual perception quality [23]. Thus, EFR’s experiments fixed  $\alpha$  at 4 and primarily adjusted  $\sigma$  to increase the level of foveation.

### 3 METHODOLOGY

#### 3.1 Overview

The goal of DEAMP is to improve the dominant-eye-aware foveated rendering method through multi-parameter optimization. Specifically, we apply two distinct sets of foveation parameters on the dominant eye and the non-dominant eye. Each eye’s visual field is divided into three nested layers based on the distance from the fixation point. Multiple foveation parameters control the level of foveation of each layer, respectively. The main objective of the multi-parameter foveation is to narrow the gap between the pixel sampling rate and the visual acuity fall-off model while ensuring perceptual quality.

The rest of this section is organized as follows. 1) We describe the used fundamental foveated rendering algorithm in Section 3.2. 2) We introduce the multi-parameter foveation in Section 3.3. 3) We finally describe the modified shading pipeline for both the dominant eye and the non-dominant eye.

#### 3.2 Foveated Rendering Technique

Our goal is to apply the foveated rendering to accelerate the deferred shading pipeline. A state-of-the-art foveated rendering technique is required as the foundation of our model. Our framework mainly consider the spatial factor into the foveated rendering. In recent years, several foveated rendering methods based on multi-spatial resolution have emerged, including KFR [23] and RMFR [45]. Meng *et al.* applied KFR on the eye-dominance-guided foveated rendering [22]. Our pilot test finds that RMFR achieves higher visual quality with less rendering artifacts and aliasing. Consequently, we chose the RMFR as the fundamental foveated rendering method.

The RMFR model contains two passes. In the first pass, RMFR downsamples the textures in the Geometry buffer (G-buffer) by the rectangular mapping image transformation. In the second pass, the transformed G-buffer undergoes a lighting pass and is then rendered to the full-resolution screen using inverse rectangular mapping.

We denote the resolution of a full-resolution G-buffer as  $W \times H$  and the resolution of the transformed G-buffer as  $w \times h$ . RMFR controls the level of foveation using a parameter  $\sigma$ , which is calculated as the ratio between the width (height) of the full-resolution G-buffer and the width (height) of the transformed G-buffer, *i.e.*:

$$\sigma = \frac{W}{w} = \frac{H}{h}. \quad (1)$$

In the G-buffer space, the foveal point  $(x_f, y_f)$  serves as a reference point for mapping each pixel’s coordinate  $(x, y)$  to  $(x', y')$ , using the following formula:

$$x' = x - x_f, \quad y' = y - y_f. \quad (2)$$

RMFR then maps the point  $(x', y')$  to  $(u, v)$  in the transformed G-buffer space through Equation (3). We take  $x' > 0, y' > 0$  as the example:

$$\begin{cases} u = \frac{1}{\sigma} \left( x_f + \frac{x'(f_x + W - x_f)}{x' + f_x} \right), \\ v = \frac{1}{\sigma} \left( y_f + \frac{y'(f_y + H - y_f)}{y' + f_y} \right), \end{cases} \quad (3)$$

where  $f_x$  and  $f_y$  are adjustable parameters for the sampling distributions in the horizontal and vertical directions, as defined by RMFR.

In the second pass, the point  $(u, v)$  is transformed back to the  $(x'', y'')$  in the G-buffer space, *i.e.*, the full-resolution screen space, by using inverse rectangular mapping presented below:

$$x'' = \frac{f_x(\sigma u - x_f)}{f_x + W - \sigma u}, \quad y'' = \frac{f_y(\sigma v - y_f)}{f_y + H - \sigma v}. \quad (4)$$

In summary,  $\sigma$  serves to regulate the resolution of the transformed G-buffer in order to accelerate rendering, while  $f_x$  and  $f_y$  are utilized to adjust the sampling distribution of different directions. These parameters can be simultaneously adjusted to approximate the visual acuity fall-off model. For instance, reducing  $f_x$  can increase the sampling rate in the center region and decrease the sampling rate in the peripheral region. However, according to Ye *et al.*’s experimental results [45], when the most optimal values of  $f_x$  and  $f_y$  are employed,  $\sigma$  can only be adjusted up to 2.6 with insignificant acceleration effects. In contrast, as indicated by our experimental results in Section 5, when using the  $f_x$  and  $f_y$  suggested by Ye *et al.* [45], the  $\sigma$  value of the central region of the dominant eye is similar to Ye *et al.*’s result ( $\sigma = 2.51$ ), whereas the  $\sigma$  value in the peripheral region can be adjusted to 4.8 in average, providing maximum acceleration.

Consequently, in our experiment, we set  $f_x = 0.38W$  and  $f_y = 0.38H$  suggested by [45]. In the following user studies, we can adjust the level of foveation by only manipulating the  $\sigma$ .

### 3.3 DEAMP Framework

The fundamental foveated rendering method in our model is explained in the last section. We further introduce the rendering framework of DEAMP. Our method could be integrated with the deferred shading pipeline, as illustrated in Fig. 2. The rasterization process stores the G-buffer of equal size to that used in normal rendering. To enhance the lighting pass for both eyes, we introduce a multi-parameter rectangular mapping and its inverse.

**Multi-Parameter (M-P) Rectangular Mapping of the G-Buffer.** For each eye, the G-buffer is divided into  $n$  eccentricity layers according to the eccentric angle from the fixation point. We define the maximum eccentric angle of the visual field as  $e^*$ . The  $n$  layers are denoted by  $L_1, L_2, \dots, L_n$ . The angular size of each layer is represented by the range  $[0, 2e_1], [2e_1, 2e_2], \dots, [2e_{n-1}, 2e^*]$ , where  $e_1 < e_2 < \dots < e_{n-1} < e^*$ .

We apply the M-P rectangular mapping to the  $n$  layers, and get  $n$  transformed G-buffers. The  $i$ -th transformed G-buffer is the output of the  $i$ -th layer of the G-buffer. The  $i$ -th transformed G-buffer is controlled by a single foveation parameter  $\sigma_i$ ,  $i = 1, 2, \dots, n$  from Equation (1). For each pixel's coordinate  $(x, y)$  in the G-buffer, we get point  $(x', y')$  relative to the foveal point by Equation (2). During our mapping, if point  $(x', y') \in L_k$ , we apply Equation (3) with  $\sigma_k$  to transform this point into the  $k$ -th transformed G-buffer.

We calculate the size of the  $k$ -th transformed G-buffer, assuming that the buffer is centered at  $(W/2, H/2)$  to simplify the condition. The eccentric angle is then transformed to the viewport distance using the projection matrix [4]. The viewport size of  $k$ -th G-buffer is defined as  $2r_{xk} \cdot 2r_{yk}$ . Therefore, the viewport size  $l_{xk} \cdot l_{yk}$  of the  $k$ -th transformed G-buffer presented below:

$$l_{xk}(f_x, \sigma_k) = \frac{2r_{xk}(f_x + \frac{W}{2})}{\sigma_k(r_{xk} + f_x)}, \quad l_{yk}(f_y, \sigma_k) = \frac{2r_{yk}(f_y + \frac{H}{2})}{\sigma_k(r_{yk} + f_y)}. \quad (5)$$

Different from the work of Guenter *et al.* [12], we discard pixels of the repeated region for further accelerating rendering. The actual viewport size of the  $k$ -th ( $k \geq 1$ ) transformed G-buffer is:

$$V_k = l_{xk} \cdot l_{yk} - \frac{\sigma_{k-1}^2}{\sigma_k^2} \cdot l_{x(k-1)} \cdot l_{y(k-1)}. \quad (6)$$

In particular, we define  $\sigma_0 = l_{x0} = l_{y0} = 0$ .

**Shading in the Nested Transformed G-Buffer.** The lighting pass computes the illumination for each pixel from different directions by utilizing information from  $n$  transformed G-buffers, respectively. The majority of rendering expenses are attributed to the lighting pass, which is directly proportional to the number of pixels being rendered [16, 37]. In comparison to the conventional deferred shading pipeline that shades  $W \times H$  pixels, while DEAMP shades a varying number of pixels for each eye, as shown below:

$$N(f_x, f_y, \sigma_1, \dots, \sigma_n) = \sum_{i=1}^n V_k. \quad (7)$$

The theoretical speedup of DEAMP for each eye can be expressed below:

$$S(f_x, f_y, \sigma_1, \dots, \sigma_n) = \frac{W \cdot H}{N(f_x, f_y, \sigma_1, \dots, \sigma_n)}. \quad (8)$$

**Inverse M-P Rectangular Mapping to the Screen.** After lighting, we apply the inverse mapping to the nested shading buffer for rendering to the screen. The final coordinate  $(x'', y'')$  in the screen is computed by Equation (4) on the  $k$ -th shading buffer.

Our mapping produces  $n$  layers  $F_1, \dots, F_n$  on the screen, the sizes of which increase gradually. We assemble them into a whole frame  $F$  applying equation (9) where  $L_0$  is defined as  $\emptyset$ :

$$F(x, y) = F_k(x, y), \quad (x, y) \in L_k \text{ and } (x, y) \notin L_{k-1}. \quad (9)$$

**The Shading Difference Between the Dominant Eye and the Non-dominant Eye.** The shading difference lies on the different sizes of the nested transformed G-buffers. Specifically, for the non-dominant eye, the rendered frame employs higher multiple levels of foveation. Therefore, the multi-parameter rectangular mapping transforms the G-buffers into smaller sizes of the transformed G-buffers compared to the output for the dominant-eye. The less number of pixels needs to render in the M-P deferred shading, saving more computational resources. The multi-parameter foveation  $\sigma_1, \dots, \sigma_n$  for each eye is measured by the user studies in Section 5.

### 3.4 System Implementation

Our DEAMP framework aims to narrow the gap between the pixel sampling rate and the visual acuity fall-off model in the identical perceptual quality. We describe the implementation details as follows.

**The Number and Sizes of the Nested Layers.** Intuitively, the G-buffer can be divided into more nested layers with the higher number of increasing  $\sigma$  according to the eccentric angle from the fixation point. However, the actual rendering time is not linear to the number of layers due to serial execution for the rendering of each layer on the GPU pipeline. There is a trade-off between the number of layers and the actual rendering time. What's more, more layers require more task time in the user study, which may cause visual fatigue. Guenter *et al.* suggested dividing the overall visual field into three eccentricity layers, with each layer's size and sampling rate chosen based on a precomputed optimization that utilized a parameter obtained through user study [12]. In our work, from the central region to periphery, the eccentric angle range of each layer is set as  $[0, 10]^\circ$ ,  $[10, 22.5]^\circ$  and  $[22.5, 45]^\circ$ , respectively. The angle ranges were used by Deng *et al.* [8].

**Transition between Layer Boundaries.** The pixel sampling rate produced by DEAMP is discontinued between two layer boundaries. However, we do not find visible borders in the context of foveated rendering using multiple parameters from user studies. We analyse the reason as follows: 1) For three eccentricity layers, our tests indicate that noticeable boundary lines will appear when the difference in foveation parameters between adjacent layers exceeds 4. However, in our user experiments, participants chose foveation parameters while guaranteeing visual quality, and data statistics indicate that the difference in foveation parameters between adjacent layers chosen by users does not exceed 2.5, as described in Section 5. 2) In our rendering framework, we incorporate Temporal Antialiasing (TAA) after merging multiple layers. TAA is a spatial anti-aliasing technique that effectively reduces flickering and aliasing artifacts in dynamic scenes. Hence, we believe it also helps mitigate the impact of boundary lines. Additionally, we apply Gaussian Blur to the periphery of the rendered images [2].

**The Rendering Engine.** We utilize the deferred shading pipeline of the Falcor engine to implement the DEAMP framework [3] on a NVIDIA GTX 1080 graphics card. The Falcor engine can report the rendering time used in each procedure. The time count regards the M-P rectangular mapping and deferred shading pass as the Pass 1, while the inverse M-P rectangular mapping is considered as the Pass 2 in Section 5. In the user studies, we apply the two passes on the Unity Engine for facilitating experimental operations, which is consistent with Meng *et al.* [22].

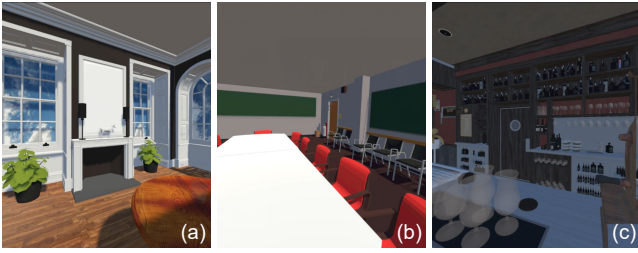


Figure 3: The scenes used in the user study. (a) *Fireplace Room* [20], (b) *Conference Room* [20], (c) *Amazon Lumberyard Bistro* [19].

#### 4 PILOT STUDY: THE COMPARISON OF RANDOM TEST AND SLIDER TEST

To evaluate the performance of DEAMP, we compare three methods in our user studies as follows. 1) We define the single-parameter foveation applied on two eyes as the 2-Layer method (2L). The 2L represents the EFR of Meng *et al.* [22], but we replace the foveated rendering method KFR with the RMFR for the consistency with our setting. 2) We define the two-parameter foveation applied on both eyes as the 4-Layer method (4L). The eccentric angle range of each layer for each eye is set as  $[0, 22.5]^\circ$  and  $[22.5, 45]^\circ$ , respectively. The 4L can also be regarded as our method. 3) Our proposed three-parameter foveation applied on both eyes is defined as the 6-Layer method (6L), which mainly represents the performance of DEAMP.

Before comparing the three aforementioned methods, it is crucial to determine if it is possible for the slider experiment to replace the random test. Both the slider test and the random test are common used experimental methods for evaluating user’s perception of foveated rendering [12, 23, 45]. In the random test, participants score the quality of foveated rendering based on different values of  $\sigma$  presented in a random order. In the slider test, participants monotonously adjust the level of foveation by themselves. However, using a random test to measure parameters in 6L would require approximately 3 hours per person with over 1200 test iterations, which could lead to visual fatigue and potentially impact the test results. In contrast, the slider test takes less time, with a testing duration of under 30 minutes for 6L.

To this end, we conduct a pilot study to ascertain whether there is significant difference between the results of the random test and the slider test. We validate it through the test of 2L. Meng *et al.* conducted an experiment to verify there was no difference between the random test and the slider test [22]. However, to ensure consistency with the settings of the main study in Section 5, we carry out another experiment, taking into account variations in experimental configuration and personnel.

##### 4.1 Apparatus

The user study is conducted in a quiet indoor room, utilizing a desktop computer with a NVIDIA GTX 1080 graphics card and an Intel Core i7-7700 CPU. Additionally, we employ HUAWEI VR Glass with a  $90^\circ$  field of view and a resolution of  $3200 \times 1600$  ( $1600 \times 1600$  per eye). This device also supports myopic refraction within a range of  $0\text{-}700^\circ$ . We use the NOLO CV1 VR controller for button interaction.

In the user study, we utilize the Unity game engine, rendering 3 distinct scenes: *Fireplace Room* [20], *Conference Room* [20] and *Amazon Lumberyard Bistro* [19], as shown in the Fig. 3. Prior to beginning the experiment, users are required to familiarize themselves fully with the tasks and interaction for a trial run using the VR devices.

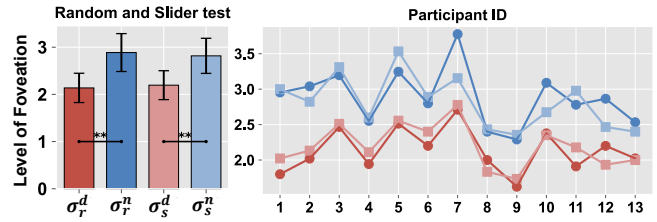


Figure 4: The values of  $\sigma$  in 2L obtained from the random test and the slider test (left). The measured parameters of 13 participants (right).

#### 4.2 Participants

For the pilot study, we recruit 13 participants (12 males and 1 female), who have normal or corrected-to-normal vision and are between the ages of 20 and 30. The individuals with myopia have degrees of myopia refraction ranging from 0 to 700, which is within the adjustment range of VR glasses mentioned previously. Majority of the participants show a good understanding of VR concepts and report prior experience to VR head-mounted displays and hand-held controllers.

#### 4.3 Design and Procedure

Participants first identify their dominant eye using the Miles Tests [30]. We conduct a slider test and a random test for 2L. Each test is conducted on three scenes mentioned before, with three trials per scene. In the random test, the participants rate the visual quality on a 5-point Likert scale referring to [22]. The experimental procedure is similar to the method used to estimate parameters in the main study of Meng *et al.* [22]. We also introduce 30% validation trials to control the impact of visual fatigue. To prevent significant changes in users’ perspectives, we conducted experiments with a fixed viewport orientation and centered fixation point, aligning with the experimental setups of KFR [23] and RMFR [45]. We increase the  $\sigma$  of the dominance eye from 1.0 to 3.0, with a step size of 0.2. In contrast, the original method decreases the  $\sigma$  [22].

#### 4.4 Results

The  $\sigma$  values of the dominant eye and the non-dominant eye in 2L are denoted as  $\sigma_r^d$  and  $\sigma_r^n$  in the random test, and as  $\sigma_s^d$  and  $\sigma_s^n$  in the slider test. We present these results in Fig. 4.

We first conducted a repeated-measure analysis of variance (RM-ANOVA) with an alpha level of 0.05 to investigate if there was a significant difference in the measured parameters between the random test and the slider test. There was no statistically significant difference between  $\sigma_r^d$  and  $\sigma_s^d$  ( $F_{1,12} = 0.229$ ,  $p = 0.637$ ), or between  $\sigma_r^n$  and  $\sigma_s^n$  ( $F_{1,12} = 0.21$ ,  $p = 0.651$ ).

To further examine the significant difference of the  $\sigma$  between both eyes in 2L, we conducted a RM-ANOVA ( $\alpha = 0.05$ ). The results of our analysis showed that the  $\sigma_r^d$  and  $\sigma_r^n$  were significantly different in the random test, as evidenced by the RM-ANOVA ( $F_{1,12} = 28.315$ ,  $p < 0.001$ ). Similarly, the RM-ANOVA conducted in the slider test demonstrated that the  $\sigma_s^d$  and  $\sigma_s^n$  were also significantly different ( $F_{1,12} = 21.623$ ,  $p < 0.001$ ). The measured parameters of each participant are shown in the right part of Fig. 4.

#### 4.5 Discussion

We discuss our results in three aspects. 1) The level of foveation of the dominant eye and the non-dominant eye differs significantly, which is consistent with the conclusion of Meng *et al.* [22]. 2) With respect to 2L test, there is no significant difference in the foveation parameters measured from the random test and the slider test. 3) To reduce the total duration of the experiment, as mentioned before, we only apply the slider test to each layer in the subsequent 4L and 6L tests. We believe that this approach would yield convincing

Table 1: The slider test for estimating the foveation parameter  $\sigma$  of each layer in  $6L$ .

Step	Adjusted $\sigma$	Highest fovea. level	Reference	Uniform change of $\sigma$
1	$\sigma_1^d$	$\hat{\sigma}_1^d$	full-resolution rendering	$\sigma_2^d = \sigma_3^d = \sigma_1^n$ $= \sigma_2^n = \sigma_3^n = \hat{\sigma}_1^d$
2	$\sigma_2^d$	$\hat{\sigma}_2^d$	frame rendered from step 1	$\sigma_3^d = \sigma_2^n =$ $\sigma_3^n = \hat{\sigma}_2^d$
3	$\sigma_3^d$	$\hat{\sigma}_3^d$	frame rendered from step 2	$\sigma_3^n = \hat{\sigma}_3^d$
4	$\sigma_1^n$	$\hat{\sigma}_1^n$	frame rendered from step 3	$\sigma_2^n = \sigma_3^n = \hat{\sigma}_1^n$
5	$\sigma_2^n$	$\hat{\sigma}_2^n$	frame rendered from step 4	$\sigma_3^n = \hat{\sigma}_2^n$
6	$\sigma_3^n$	$\hat{\sigma}_3^n$	frame rendered from step 5	—

results since the slider test of  $4L$  or  $6L$  can be divided into two or three independent slider tests of  $2L$ . Therefore, we only employ the slider test in the main study.

We anticipate that the issues encountered in this experiment can provide valuable insights for future researchers. We conducted a total of two pilot studies. In the first study, our experimental findings indicated that there were no significant differences between the  $\sigma$  values for the dominant and non-dominant eye; these values were similar. We attribute this outcome to a flaw in our first experimental setup. The first and the final experimental setups differ primarily in two aspects: 1) Participants were allowed to freely rotate their heads to observe the scene during the first experiment, whereas in the final experiment, head direction was fixed. 2) In order to mitigate the potential impact of estimating the participants' gaze, we employed a stimulus-guided approach to determine the position of the fixation point in the first experiment. Participants' fixation points moved randomly within the screen area and were centered on a digit that underwent random color changes. In the final experiment, participants were required to fixate on the center of the screen. These experimental differences may result in variations in individuals' head rotations and eye movements when assessing different layer parameters in dynamic scenes. It can lead to different perceptions of object edge flickering. Consequently, this introduces additional variables to the parameter evaluation, making it challenging to draw definitive conclusions.

## 5 MAIN STUDY: ESTIMATION OF FOVEATION PARAMETERS FOR $2L$ , $4L$ AND $6L$

As stated in Section 4, the primary objective of this study is to assess the rendering efficiency of our proposed methods ( $4L$  and  $6L$ ) compared to other foveated rendering methods, *e.g.*,  $2L$ , while maintaining perceptual quality. Specifically, we aim to identify which particular layer of a foveated rendering frame show the best accelerating performance. For each layer in these methods, we measure the acceptable foveation parameter  $\sigma$  through a slider test. This enables us to perform the dominant-eye-aware foveated rendering with minimal perceptual differences, based on the measured  $\sigma$  for every layer.

### 5.1 Participants

We recruit 16 university students (14 males and 2 females), aged between 20 to 30 years old (mean = 24) for the main study, who have normal or corrected-to-normal vision. Among them, 12 participants are right-eye dominant, while 4 participants are left-eye dominant. We used the identical apparatus to the one employed in the pilot study.

## 5.2 Design and Procedure

We conduct three tests in the main study, namely  $2L$  test,  $4L$  test and  $6L$  test. In among tests, the goal is to find the highest  $\sigma$  of each layer in  $2L$ ,  $4L$  and  $6L$ , which still produces an acceptable level of foveation for both eyes. Due to the change in the experimental personnel, we repeat the  $2L$  test using the same slider test as the pilot study. We introduce the  $4L$  test and  $6L$  test below. The  $4L$  test shares a similar design with the  $6L$  test, and thus we do not elaborate further.

**6L test:** In the  $6L$  test, we conduct the slider test on each layer. The order of layer adjustment follows the principle of "inner before outer, dominance before non-dominance". We denote the foveation parameters of  $6L$  as  $\sigma_1^d, \sigma_2^d, \sigma_3^d, \sigma_1^n, \sigma_2^n$  and  $\sigma_3^n$ , with all values set to 1.0 initially. These values increase with a step size of 0.2. The maximum value of  $\sigma$  is set as 8.0 and got from our preliminary test. This  $6L$  test contains six steps corresponding to the estimations of six  $\sigma$  values, as shown in Tab.1. In Step 1, participants increase the  $\sigma_1^d$  progressively. The goal is to identify the highest level of foveation, denoted as  $\hat{\sigma}_1^d$ , that is perceptually equivalent to the reference with full-resolution rendering. We express this condition as  $\sigma_2^d = \sigma_3^d = \sigma_1^n = \sigma_2^n = \sigma_3^n = \hat{\sigma}_1^d$ . Please refer to the Tab.1 for steps 2-5.

**Task Procedure:** In the pilot study, we observe that the mean difference in  $\sigma$  between layers of each eye remained stable ( $\pm 0.2$ ) across three trials per scene. Consequently, to minimize the experimental duration, we reduce the number of trials per scene from three to two. Participants are first provided with a brief introduction to the test procedures for  $2L$ ,  $4L$  and  $6L$  test. To help participants become familiar with all methods, we include a warm-up trial using the same scenes. Next, participants complete formal experiments which involved completing the three tests in a counterbalanced order using a Latin Square design. The entire experiment lasted approximately 60 minutes.

## 5.3 Evaluation Metrics

**Performance comparison of  $2L$ ,  $4L$  and  $6L$ :** we define three objective metrics to evaluate the performance of participants for different methods.

- **Level of Foveation:** participants' selection of the  $\sigma$  value for each layer during the three tests. This information can directly compare the level of foveation between  $2L$ ,  $4L$  and  $6L$ .
- **Total Rendered Pixels Count:** in the lighting pass, the number of pixels rendered of  $2L$ ,  $4L$  and  $6L$ . More pixels count means longer rendering time.
- **Theoretical Speedup:** the total number of display pixels divided by the total rendered pixels count. Higher speedup indicates less rendering time.
- **Rendering Time:** the rendering time of different methods using the deferred shading pipeline. This enables a direct comparison of the rendering acceleration.

**Evaluation of the accelerating effect from inner to outer, from dominance to non-dominance:** besides the above overall analysis, we further define two objective metrics to identify which particular layer accelerates the rendering most notably, and the rendered frame of which eye provides more acceleration.

- **Accumulative Pixel Distribution:** proportion of the number of rendered pixels to the total number of display pixels as eccentricity increases. This metric aim to explore the accelerating effect of each region from inner to outer regions.

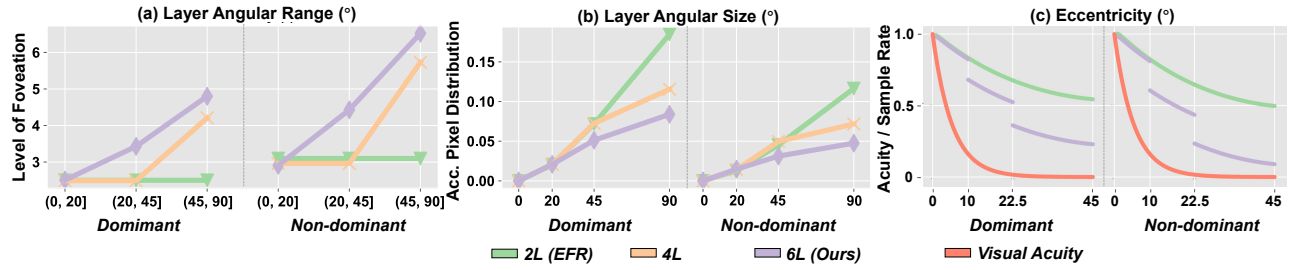


Figure 5: Line charts of the foveated rendering methods' performance under different measurements. The comparison of Level of Foveation (a), the Accumulative Pixel Distribution (b), and the variation of the sampling rate and human visual acuity (c).

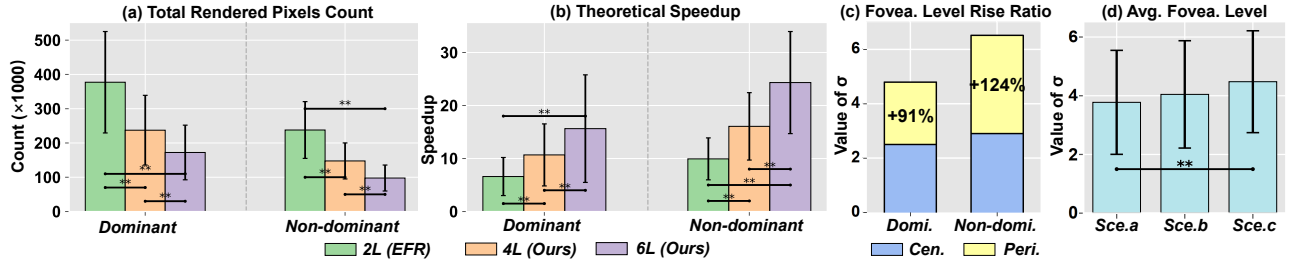


Figure 6: Bar charts of the foveated rendering methods' performance under different measurements. Error bars indicate the standard error. The statistical significances are labeled with \*\* ( $p < 0.05$ ).

- **Foveation Level Rise Ratio:** the comparison of increase ratio in foveation level from inner to outer layers between the dominant and non-dominant eyes. This metric intuitively describes the speedup effect of different eyes.

**Evaluation of the results of approximation for visual acuity modeling:** we define one objective metric to validate the efficacy for our approach.

- **Acuity/Sampling Rate:** the comparison between the visual acuity fall-off model with decreasing sampling rate normalized in the  $2L$  and  $6L$  with the identical perceptual quality.

**Evaluation of the differences in visual perception among different scenes:** we define one objective metric to evaluate the impact of different scenes on the foveation level of  $2L$ ,  $4L$  and  $6L$ .

- **Average Foveation Level:** the comparison of the average foveation level between different scenes, and the analysis of the foveation level for individual scene.

## 5.4 Results and Discussion

We conducted RM-ANOVAs ( $\alpha = 0.05$ ) and post hoc pairwise  $t$ -tests to judge whether a certain metric is significantly different across different input.

The performance comparisons of  $2L$ ,  $4L$  and  $6L$  are shown in Fig. 5 and Fig. 6. We first analyzed the Level of Foveation for different methods, as shown in Fig. 5(a), from which we make the following observations. 1) The average level of foveation across all methods within the range of  $[0, 20]^\circ$  is notably consistent. 2) The layers with the range of  $[20, 45]^\circ$  in  $6L$  can accommodate a higher level of foveation compared to both  $2L$  and  $4L$ . 3) For the range of  $[45, 90]^\circ$ , the foveation level of the  $4L$  and  $6L$  has reached more aggressive level, but there exists a small gap. We argue that by increasing the foveation level of the range of  $[20, 45]^\circ$ , the threshold for acceptable foveation level in the range of  $[45, 90]^\circ$  may increase accordingly. Two RM-ANOVAs demonstrated that different methods

had a significant difference on the Total Rendered Pixels Count ( $F_{2,30} = 13.641$ ,  $p < 0.001$  for the dominant eye;  $F_{2,30} = 21.887$ ,  $p < 0.001$  for the non-dominant eye), as shown in Fig. 6(a). A post hoc test revealed that the  $6L$  rendered significantly fewer pixels than the other divisions for the dominant eye ( $6L-4L$ ,  $p < 0.001$ ;  $6L-2L$ ,  $p < 0.001$ ). The non-dominant eye exhibited a similar performance. Two RM-ANOVAs showed that different divisions had a significant main effect on Theoretical Speedup ( $F_{2,30} = 6.548$ ,  $p = 0.003$  for the dominant eye;  $F_{2,30} = 16.832$ ,  $p < 0.001$  for the non-dominant eye), as shown in Fig. 6(b). A post hoc test indicated that the  $6R$  division achieved significantly higher speedup than the other divisions for the dominant eye ( $6L-4L$ ,  $p = 0.001$ ;  $6L-2L$ ,  $p < 0.001$ ). Similar results were found for the non-dominant eye. The Rendering Time of different methods is shown in Tab. 2. We found the  $6L$  achieved  $3.4\times$  speedup compared to the full-resolution rendering, while the EFR achieved  $2.9\times$  speedup. We have also conducted a comparison of alternative methods, which can be found in Tab. 2.

Next, we evaluated the impact of accelerating on different layers. Fig. 5(b) displays the results of the Accumulative Pixel Distribution. From our findings, there was minimal variation in the number of rendered pixels among different methods for layer sizes ranging from  $0^\circ$  to  $45^\circ$ . In contrast, layer sizes between  $45^\circ$  and  $90^\circ$  showed a rapid increase in the number of rendered pixels for the  $2L$ , while the  $6L$  had a slower increase. This can be attributed to the fact that, in a full-resolution frame, the outer layer comprises 75% of the total rendered pixel count, thus making it crucial to expedite rendering in this layer. As a result, employing an aggressive yet acceptable foveation level on the periphery led to fewer pixels being rendered in the  $6L$ . Consequently, attaining a substantial speedup relies on the optimization of the outer layer of the frame. The Foveation Level Rise Ratio is illustrated in Fig. 6(c). Noticeably, the foveation level in the peripheral region has substantially increased when compared to the center, with greater enhancement for the non-dominant eye. In conclusion, it is crucial to improve the foveation level within the peripheral region of  $45$  to  $90^\circ$  for enhancing rendering speed without compromising visual perception quality, especially in the non-dominant eye's periphery.

Table 2: The rendering time and frame-rates (in Fps) comparison among full-resolution rendering, KFR [23], RMFR [45], EFR [22], 2R [22], 4R (Ours) and 6R (Ours) on binocular screens with a resolution of 1600×1600 per eye.

Procedure (ms)	Full-resolution	KFR [23]	RMFR [45]	EFR (KFR) [22]		2R (EFR) [22]		4R (Ours)		6R (Ours)	
				dom	n-dom	dom	n-dom	dom	n-dom	dom	n-dom
Depth Pass	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20
Shadow Pass	0.35	0.35	0.35	0.35	0.35	0.35	0.35	0.34	0.34	0.35	0.35
Defer Pass	3.93	4.05	4.11	4.08	4.03	4.13	4.09	3.99	3.94	<b>3.89</b>	3.92
Skybox	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Shading/Pass1	19.27	5.49	3.34	4.52	2.19	3.34	2.53	2.94	2.06	2.76	<b>1.97</b>
Pass2	/	0.28	0.22	0.25	0.23	0.22	0.23	0.11	0.12	0.13	<b>0.10</b>
TAA	0.65	0.43	<b>0.42</b>	0.43	0.43	<b>0.42</b>	<b>0.42</b>	0.48	0.48	0.43	<b>0.42</b>
Total GPU Time	24.41	10.81	8.65	9.84	7.44	8.67	7.83	8.07	7.15	7.77	<b>6.97</b>
Binocular Time (ms)	48.82	21.62	17.30	17.28		16.50		15.22		<b>14.74</b>	
Fps	20	46	58	58		61		66		<b>68</b>	

We evaluated the efficacy of approximation for visual acuity modeling. The Fig. 5(c) showed the comparison between the visual acuity fall-off model and the decrease of the normalized sampling rate of  $2L$  and  $6L$ . The average foveation level measured in the user study was used as the value of  $\sigma$  for  $2L$  and  $6L$ . We found the  $2L$  applied a single-parameter foveation to the entire visual field of each eye, which created a significant gap between the pixel sampling rate and the visual acuity fall-off model, leaving ample room for further optimization in the periphery. In contrast,  $6L$  used multi-parameter foveation to control the levels of foveation for the entire image of each eye. As the eccentricity increased, the sampling rate approached the visual acuity fall-off model more closely when compared to the line of  $2L$ . The non-dominant eye was found to be closer to the visual acuity fall-off model than the dominant eye. Hence, we concluded that the  $6L$  can be effectively used for visual acuity modeling.

Finally, we evaluate the differences in visual perception among different scenes. The RM-ANOVA showed that different scenes had a significant difference on the level of foveation of each person ( $F_{2,30} = 3.817, p = 0.023$ ). A post hoc test revealed that the level of foveation of Scene (c) was higher than that of Scene (a) ( $p = 0.018$ ). The difference of foveation parameters in difference scenes also exists in Ye *et al.* [45]. Although the difference exists, we found the significant difference between  $\sigma^d$  and  $\sigma^n$  of the dominant eye and the non-dominant eye was consistent for Scene (a) and Scene (c). Moreover, the significant difference between the pairwise values of  $\sigma_1, \sigma_2$  and  $\sigma_3$  in the foveal, middle and peripheral layers remained consistent for these two scenes. Therefore, to facilitate our presentation, we considered the three scenarios as a whole, as previously mentioned.

## 5.5 Summary of Key Findings in Main Study

Based on the above results and analysis of the main study, several key findings can be summarized as follows:

- Our designed DEAMP ( $6L$ ) outperforms other foveated rendering methods in terms of speedup. In particular, our method significantly reduces the number of pixels shaded during the lighting pass.
- Enhancing the level of foveation in the peripheral region, particularly in the non-dominant eye’s periphery, is vital for substantially improving rendering speed.
- The DEAMP effectively minimizes the disparity between the pixel sampling rate and the visual acuity fall-off model in consistent perceptual quality.

## 6 LIMITATIONS AND FUTURE WORK

The theoretical speedup is inversely proportional to the number of pixels rendered. The actual rendering time is not totally matching

the proportion. This is because we found it is difficult to render multiple nested layers in parallel on the GPU pipeline due to the limitation of the Falcor framework. In the future, we will explore the parallel rendering method for multiple nested layers, to further improve the rendering speed.

As our study primarily focused on perceptual experiments, some users occasionally experienced visual fatigue. In such instances, we prompted them to take breaks as needed. In future work, we plan to conduct experiments in dynamic scenes to explore the potential negative impacts of the dominant eye theory. Furthermore, the results indicate considerable variation in the level of foveation observed among participants in each scene. To effectively apply this method in real-world scenarios, we recommend employing foveation parameters that are universally acceptable for all scenes and participants.

The pilot study and main study each lasted approximately 60 minutes. In the future, we plan to design an algorithm to reduce the time for parameter selection. For instance, in the slider test, we aim to use the binary search method to decrease the number of selection iterations from  $N$  to  $\log N$ . Furthermore, eye dominance can affect the decomposition of the layers, such as the number of layers and the eccentricity angle. We plan to optimize both the layer decomposition and the foveation parameters simultaneously for optimal results.

In the future, we will investigate the feature of human binocular vision in a complete different manner. The different number of divided layers may be applied to the dominant eye and the non-dominant eye, respectively. What’s more, the recent work explored the foveated rendering in augmented reality [15]. We argue that the theory of eye dominance may have a positive impact on rendering in AR, especially on mobile AR devices.

## 7 CONCLUSION

In this research, we proposed the dominant-eye-aware foveated rendering optimized with multi-parameter foveation (DEAMP). Concretely, we controlled the level of foveation for the dominant eye and the non-dominant eye with two distinct sets of foveation parameters. Each eye’s visual field was divided into three nested layers based on the eccentric angle from the fixation point. Multiple parameters governed the level of foveation of each layer, respectively. We conducted user studies to evaluate our method, and the experimental results demonstrate that DEAMP provides superior rendering times while reducing the disparity between pixel sampling rate and the visual acuity fall-off model, all while maintaining perceptual quality.

## ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant 62372019 and the Academic Excellence Foundation of BUAA for PhD Students. We are thankful to Jiannan Ye and Xiaoxu Meng for providing source code and kindly help.



## REFERENCES

- [1] Computer Graphics. [Online]. <https://www.cs.utexas.edu/~theshark/courses/cs354/>, Accessed June 1, 2023.
- [2] Gaussian blur. [Online]. [https://en.wikipedia.org/wiki/Gaussian\\_blur](https://en.wikipedia.org/wiki/Gaussian_blur), Accessed June 12, 2023.
- [3] The Falcor rendering framework. [Online]. <https://github.com/NVIDIAGameWorks/Falcor>, Accessed June 12, 2023.
- [4] Unity. [Online]. <https://docs.unity3d.com/ScriptReference/Camera.WorldToViewportPoint.html>, Accessed June 12, 2023.
- [5] R. Albert, A. Patney, D. Luebke, and J. Kim. Latency Requirements for Foveated Rendering in Virtual Reality. *ACM Trans. Appl. Percept.*, 14(4), sep 2017. doi: 10.1145/3127589
- [6] Y. Bao, J. Wang, Z. Wang, and F. Lu. Exploring 3d interaction with gaze guidance in augmented reality. In *2023 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pp. 22–32, 2023. doi: 10.1109/VR55154.2023.00018
- [7] B. Chaurasia and B. Mathur. Eyedness. *Acta Anatomica*, 96(2):301–305, 07 2008. doi: 10.1159/000144681
- [8] N. Deng, Z. He, J. Ye, B. Duinkharjav, P. Chakravarthula, X. Yang, and Q. Sun. FoV-NeRF: Foveated Neural Radiance Fields for Virtual Reality. *IEEE Transactions on Visualization and Computer Graphics*, 28(11):3854–3864, 2022. doi: 10.1109/TVCG.2022.3203102
- [9] A. Deza and T. Konkle. Emergent Properties of Foveated Perceptual Systems. *CoRR*, abs/2006.07991, 2020.
- [10] L. Franke, L. Fink, J. Martschinke, K. Selgrad, and M. Stamminger. Time-Warped Foveated Rendering for Virtual Reality Headsets. *Computer Graphics Forum*, 40(1):110–123, 2021. doi: 10.1111/cgf.14176
- [11] M. Fujita and T. Harada. Foveated real-time ray tracing for virtual reality headset. *Light Transport Entertainment Research*, 2014.
- [12] B. Guenter, M. Finch, S. Drucker, D. Tan, and J. Snyder. Foveated 3D Graphics. *ACM Trans. Graph.*, 31(6), nov 2012. doi: 10.1145/2366145.2366183
- [13] S. Jabbireddy, X. Sun, X. Meng, and A. Varshney. Foveated rendering: Motivation, taxonomy, and research directions. *arXiv preprint arXiv:2205.04529*, 2022.
- [14] A. S. Kaplanyan, A. Sochenov, T. Leimkühler, M. Okunev, T. Goodall, and G. Rufo. DeepFovea: Neural Reconstruction for Foveated Rendering and Video Compression Using Learned Statistics of Natural Videos. *ACM Trans. Graph.*, 38(6), nov 2019. doi: 10.1145/3355089.3356557
- [15] J. Kim, Y. Jeong, M. Stengel, K. Akşit, R. Albert, B. Boudaoud, T. Greer, J. Kim, W. Lopes, Z. Majercik, P. Shirley, J. Spjut, M. McGuire, and D. Luebke. Foveated AR: Dynamically-Foveated Augmented Reality Display. *ACM Trans. Graph.*, 38(4), jul 2019. doi: 10.1145/3306346.3322987
- [16] M. Koskela, A. Lotvonen, M. Mäkitalo, P. Kivi, T. Viitanen, and P. Jääskeläinen. Foveated Real-Time Path Tracing in Visual-Polar Space. 2019. doi: 10.2312/sr.20191219
- [17] M. Koskela, T. Viitanen, P. Jääskeläinen, and J. Takala. Foveated Path Tracing. In *Advances in Visual Computing*, pp. 723–732. Springer International Publishing, Cham, 2016.
- [18] B. Koçtekin, N. Ünay Gündoğan, A. G. K. Altıntaş, and A. C. Yazıcı. Relation of eye dominance with color vision discrimination performance ability in normal subjects. *Int J Ophthalmol*, 6(15):733–8, 2013 Oct. doi: 10.3980/j.issn.2222-3959.2013.05.34
- [19] A. Lumberyard. Amazon Lumberyard Bistro, Open Research Content Archive (ORCA), July 2017. <http://developer.nvidia.com/orca/amazon-lumberyard-bistro>.
- [20] M. McGuire. Computer Graphics Archive, July 2017. <https://casual-effects.com/data>.
- [21] X. Meng, R. Du, J. F. JaJa, and A. Varshney. 3D-kernel foveated rendering for light fields. *IEEE Transactions on Visualization and Computer Graphics*, 27(8):3350–3360, 2020.
- [22] X. Meng, R. Du, and A. Varshney. Eye-dominance-guided Foveated Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 26(5):1972–1980, 2020. doi: 10.1109/TVCG.2020.2973442
- [23] X. Meng, R. Du, M. Zwicker, and A. Varshney. Kernel Foveated Rendering. *Proc. ACM Comput. Graph. Interact. Tech.*, 1(1), jul 2018. doi: 10.1145/3203199
- [24] A. Oishi, S. Tobimatsu, K. Arakawa, T. Taniwaki, and J. ichi Kira. Ocular dominance in conjugate eye movements at reading distance. *Neurosci Res.*, 52(3):263–8, 2005 Jul. doi: 10.1016/j.neures.2005.03.013
- [25] A. Patney, J. Kim, M. Salvi, A. Kaplanyan, C. Wyman, N. Benty, A. Lefohn, and D. Luebke. Perceptually-based foveated virtual reality. In *ACM SIGGRAPH 2016 emerging technologies*, pp. 1–2. 2016.
- [26] A. Patney, M. Salvi, J. Kim, A. Kaplanyan, C. Wyman, N. Benty, D. Luebke, and A. Lefohn. Towards Foveated Rendering for Gaze-Tracked Virtual Reality. *ACM Trans. Graph.*, 35(6), dec 2016. doi: 10.1145/2980179.2980246
- [27] A. Patney, M. Salvi, J. Kim, A. Kaplanyan, C. Wyman, N. Benty, D. Luebke, and A. Lefohn. Towards Foveated Rendering for Gaze-Tracked Virtual Reality. *ACM Trans. Graph.*, 35(6), dec 2016. doi: 10.1145/2980179.2980246
- [28] M. Pharr, C. Kolb, R. Gershbein, and P. Hanrahan. Rendering complex scenes with memory-coherent ray tracing. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pp. 101–108, 1997.
- [29] C. Porac and S. Coren. The dominant eye. *Psychological bulletin*, 83(5):880, 1976.
- [30] H. L. Roth, A. N. Lora, and K. M. Heilman. Effects of monocular viewing and eye dominance on spatial attention. *Brain*, 125(9):2023–2035, 09 2002. doi: 10.1093/brain/awf210
- [31] X. Shi, L. Wang, X. Wei, and L.-Q. Yan. Foveated Photon Mapping. *IEEE Transactions on Visualization and Computer Graphics*, 27(11):4183–4193, 2021. doi: 10.1109/TVCG.2021.3106488
- [32] E. Shneur and S. Hochstein. Eye dominance effects in conjunction search. *Vision Research*, 48(15):1592–1602, 2008. doi: 10.1016/j.visres.2008.04.021
- [33] C. Soler, O. Hoel, and F. Rochet. A Deferred Shading Pipeline for Real-Time Indirect Illumination. In *ACM SIGGRAPH 2010 Talks, SIGGRAPH '10*. Association for Computing Machinery, New York, NY, USA, 2010. doi: 10.1145/1837026.1837049
- [34] M. Stengel, S. Grogorick, M. Eisemann, and M. Magnor. Adaptive Image-Space Sampling for Gaze-Contingent Real-Time Rendering. *Comput. Graph. Forum*, 35(4):129–139, jul 2016.
- [35] T. Tariq, C. Tursun, and P. Didyk. Noise-based enhancement for foveated rendering. *ACM Transactions on Graphics (TOG)*, 41(4):1–14, 2022.
- [36] E. Turner, H. Jiang, D. Saint-Macary, and B. Bastani. Phase-Aligned Foveated Rendering for Virtual Reality Headsets. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 1–2, 2018. doi: 10.1109/VR.2018.8446142
- [37] O. T. Tursun, E. Arabadzhyska-Koleva, M. Wernikowski, R. Mantiuk, H.-P. Seidel, K. Myszkowski, and P. Didyk. Luminance-Contrast-Aware Foveated Rendering. *ACM Trans. Graph.*, 38(4), jul 2019. doi: 10.1145/3306346.3322985
- [38] I. Wald, P. Slusallek, C. Benthin, and M. Wagner. Interactive rendering with coherent ray tracing. In *Computer graphics forum*, vol. 20, pp. 153–165. Wiley Online Library, 2001.
- [39] L. Wang, X. Shi, and Y. Liu. Foveated rendering: A state-of-the-art survey. *Computational Visual Media*, 9(2):195–228, 2023.
- [40] Z. Wang, H. Wang, H. Yu, and F. Lu. Interaction with gaze, gesture, and speech in a flexibly configurable augmented reality system. *IEEE Transactions on Human-Machine Systems*, 51(5):524–534, 2021.
- [41] Z. Wang, H. Yu, H. Wang, Z. Wang, and F. Lu. Comparing single-modal and multimodal interaction in an augmented reality system. In *2020 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pp. 165–166. IEEE, 2020.
- [42] Z. Wang, Y. Zhao, and F. Lu. Control with vergence eye movement in augmented reality see-through vision. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops*, pp. 548–549. IEEE, 2022.
- [43] Z. Wang, Y. Zhao, and F. Lu. Gaze-Vergence-Controlled See-Through Vision in Augmented Reality. *IEEE Transactions on Visualization and Computer Graphics*, 28(11):3843–3853, 2022. doi: 10.1109/TVCG.2022.3203110
- [44] M. Weier, T. Roth, E. Kruijff, A. Hinkenjann, A. Pérard-Gayot, P. Slusallek, and Y. Li. Foveated Real-Time Ray Tracing for Head-

Mounted Displays. *Computer Graphics Forum*, 35(7):289–298, 2016. doi: 10.1111/cgf.13026

- [45] J. Ye, A. Xie, S. Jabbireddy, Y. Li, X. Yang, and X. Meng. Rectangular Mapping-based Foveated Rendering. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 756–764, 2022. doi: 10.1109/VR51125.2022.00097