

Dominant-Eye-Aware Asymmetric Foveated Rendering for Virtual Reality

Zhimin Wang, Xiangyuan Gu, and Feng Lu, *Senior Member, IEEE*

Abstract—To address the increasing computational demands of high-resolution virtual reality headsets, foveated rendering reduces pixel sampling in the peripheral regions of the visual field. However, existing methods have not fully leveraged binocular vision, particularly the dominant eye theory. In our prior work, we proposed Dominant-Eye-Aware foveated rendering optimized with Multi-Parameter foveation (DEAMP), which divided each eye’s visual field into fixed eccentricity layers ($[0, 10]^\circ$, $[10, 22.5]^\circ$, $[22.5, 45]^\circ$). The non-dominant eye received greater foveation within the same layers compared to the dominant eye. We further argue that the eccentricity ranges should vary between the eyes due to inter-eye, individual, and scene-specific differences. In this article, we introduce an enhanced method, Dominant-Eye-aware Asymmetric Foveated Rendering (DEA-FoR). This treats eccentricity as a new variable, allowing users to select eccentricity sets tailored to their eyes and supporting asymmetric configurations between the two eyes. Experimental results demonstrate significant improvements in rendering speed over our previous method while maintaining perceptual quality. Additionally, we found that individual differences and scene texture complexity significantly influence the eccentricity settings. This work offers new insights into perceptual differences in binocular vision and contributes to optimizing virtual reality experiences. Our project page is at https://zhimin-wang.github.io/DEA_FoR.html.

Index Terms—Virtual reality, foveated rendering, eye dominance, binocular vision perception.

I. INTRODUCTION

VIRTUAL reality (VR) creates immersive environments through advanced computer graphics. With the rise of the metaverse, VR has become a versatile platform with applications in entertainment, online education, and healthcare [1]–[5]. Despite its potential, the significant computational demands of rendering high-quality VR content remain a major barrier to its widespread adoption [6], [7]. To address this, researchers are exploring techniques to reduce rendering overhead and improve VR accessibility.

Foveated rendering offers an effective solution by utilizing the characteristics of the human visual system. The fovea, a region rich in cones, enables detailed vision and is essential for high-resolution tasks. In contrast, the peripheral retina, with sparser cones, perceives less detail [8], [9]. By rendering high-quality visuals in the foveal region and lower-quality visuals in peripheral areas, foveated rendering minimizes computation without sacrificing perceptual quality. To implement foveated

Zhimin Wang, Xiangyuan Gu, and Feng Lu are with the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing 100191, China. e-mail: {zmn.wang \ guxy \ lufeng}@buaa.edu.cn.

Feng Lu is the corresponding author.

Manuscript received January 10, 2025.

rendering, current research focuses on two techniques: rasterization and ray tracing [6], [10], [11]. In rasterization, foveated approaches reduce the resolution of geometric buffers (G-buffers) and decrease sampling rates in peripheral regions [12]–[15]. In ray tracing, computational efficiency is achieved by casting fewer rays in peripheral areas [16]–[18].

While these methods improve efficiency, they have not fully exploited binocular vision, particularly the dominant eye theory. The dominant eye exhibits higher visual acuity and color discrimination than the non-dominant eye [19]–[21], with about 70% of individuals being right-eye dominant and 29% left-eye dominant [22]. Leveraging this, researchers can render images at different resolutions for each eye to reduce computation. For example, Meng *et al.* [23] apply higher foveation levels to the non-dominant eye, achieving greater acceleration than traditional methods (see Fig. 1 (a)). In this paper, the “foveation level” represents the overall degree of blur from center to periphery. However, their single-parameter model poorly matches the visual acuity fall-off model in peripheral regions, causing unnecessary rendering costs, as shown by the purple and green lines in Fig. 1 (d)(e). Thus, a more refined integration of foveated rendering and dominant eye theory is needed to further optimize VR rendering.

To address this challenge, in our previous work [24], we proposed a Dominant-Eye-Aware foveated rendering method, optimized through Multi-Parameter foveation (DEAMP). This approach employs distinct foveation parameters for the dominant and non-dominant eyes, as shown in Fig. 1 (b). For each eye, the visual field is divided into three eccentricity layers based on angular distance from the fixation point. Within each layer, single-parameter foveation is applied to modulate the level of foveation. Combining these layers, multi-parameter foveation controls the overall rendering quality for each eye’s image. The objective is to align the pixel sampling rate with the visual acuity fall-off model while ensuring identical perceptual quality. Results show that our method significantly reduces rendering time compared to existing approaches.

In our previous work [24], the eccentricity range for each layer is fixed, *i.e.*, $[0, 10]^\circ$, $[10, 22.5]^\circ$, and $[22.5, 45]^\circ$. While this approach significantly reduces rendering time, we recognize the need to further improve the DEAMP by exploring asymmetric eccentricity layer optimization for both eyes. This is motivated by two factors: 1) The dominant eye exhibits higher visual acuity than the non-dominant eye, suggesting that the optimal eccentricity range for each layer may vary between the two eyes. 2) Our previous study [24] revealed significant variations in rendering acceleration across different scenes and individuals, indicating that the optimal eccentricity

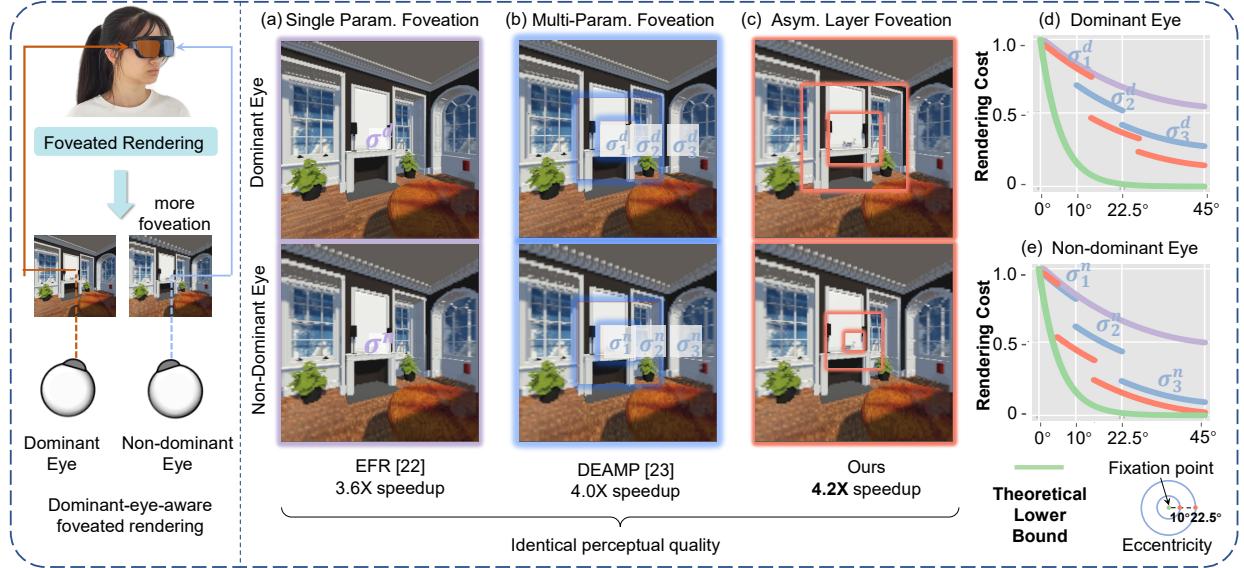


Fig. 1: To enhance rendering efficiency, we propose DEA-FoR, a Dominant-Eye-aware Asymmetric Foveated Rendering method. It achieves a **4.2 \times** speedup compared to full-resolution rendering, while its **rendering cost** (pixel sampling rate) remains minimal and approaches the **theoretical lower bound** (human visual acuity). Evaluations are conducted on binocular screens with a resolution of 1600 \times 1600 per eye.

layers may vary under different conditions. Thus, introducing eccentricity as a new variable is necessary.

Therefore, in this paper, extended from the DEAMP, we propose the Dominant-Eye-aware Asymmetric Foveated Rendering method (DEA-FoR). This allows users to customize eccentricity sets based on their eyes' traits and enables asymmetric eccentricity layer partitioning between the dominant and non-dominant eyes, as shown in Fig. 1 (c). This represents a significant advancement and distinction from the earlier conference version [24]. DEA-FoR retains the three-layer structure for each eye's visual field. It optimizes eccentricity sets to adjust layer sizes and applies multi-parameter foveation to regulate foveation levels. To accelerate foveation parameter collection in the user study, we propose a more efficient parameter selection method compared to DEAMP [24]. Experimental results show that DEA-FoR achieves substantial improvements in rendering speed while maintaining perceptual quality. Additionally, individual differences and scene texture complexity significantly affect the eccentricity settings. This work offers valuable insights into binocular vision and provides practical guidance for optimizing VR experiences.

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

- We propose DEA-FoR, a dominant-eye-aware asymmetric foveated rendering method that jointly optimizes eccentricity sets and foveation parameters, significantly improving rendering speed while preserving visual quality.
- We introduce a more efficient parameter selection method for user-specific eccentricity sets and foveation parameters. This method reduces the experiment duration from three hours to just 20 minutes.
- We conduct user studies with 22 participants across three scenes. The results demonstrate the effectiveness of DEA-FoR and reveal that individual differences and scene texture complexity significantly influence the eccentricity and foveation parameter settings.

- Our method uses a GPU-based deferred shading pipeline and achieves a 4.2 \times speedup over full-resolution rendering at 1600 \times 1600 pixels per eye.

II. RELATED WORK

In this section, we review recent advancements in foveated rendering for VR and examine the effect of eye dominance on rendering.

A. Foveated Rendering

Foveated rendering techniques utilize principles of the human visual system to reduce computational cost in rendering high-quality VR content while preserving perceptual quality for users [9], [25], [26]. This subsection provides an overview of the human visual system, followed by an analysis of the main categories of foveated rendering techniques.

Human visual system. Foveated rendering builds on the trait of the human visual system [6]. Visual acuity drops by over 95% from central to peripheral vision [6], [27]. Foveated rendering exploits this by applying high-quality rendering to the foveal region and low-quality rendering to the periphery, significantly reducing rendering load. Krajancich *et al.* [28] further applied lower frame rates to the visual periphery while maintaining perceived visual quality.

Rasterization. Foveated rendering, as reviewed by Wang *et al.* [6], commonly includes rasterization, ray tracing and other methods. In deferred shading pipelines, rasterization stores geometric information in G-buffers and computes per-pixel lighting. To accelerate rendering, existing techniques reduce G-buffer resolution and lower peripheral sampling rates. Guenter *et al.* [12] proposed methods that segment the visual field into multiple layers with varying sampling rates based on eccentricity, though they do not account for the dominant eye effect. Meng *et al.* [13] presented a two-pass kernel foveated

rendering (KFR) pipeline, using kernel log-polar transformation for downsampling the G-buffer in the first pass and inverse transformation for upsampling the reduced-resolution buffer in the second. Ye *et al.* [14] adopted a similar two-pass design but employed Rectangle-Mapped Foveated Rendering (RMFR) to achieve finer foveation with fewer artifacts. Fan *et al.* [29] developed a scene-aware foveated rendering approach that combines scene features, visual acuity, and gaze motion to enhance peripheral quality. Some researchers also incorporated temporal factors into foveated rendering [9], [15], [30].

Ray tracing rendering. Ray tracing adjusts the number of rays projected onto each pixel to balance visual quality and computational cost [31], [32]. Higher ray counts improve fidelity but increase overhead. Foveated rendering accelerates this process by reducing rays in peripheral regions of the visual field. Koskela *et al.* [33] introduced a visual-polar space for foveated ray tracing to further minimize ray requirements.

Other foveated rendering. In addition to the previously mentioned rendering paradigms, various studies have explored alternative approaches to achieve foveated rendering. For example, recent advances in neural rendering [34], [35] have extended foveated rendering techniques by incorporating gaze-contingent neural representations, demonstrating that neural radiance fields can be adapted to exploit human visual perception for efficient VR rendering.

B. Eye Dominance

The concept of eye dominance has recently been utilized to accelerate rendering. Eye dominance, a characteristic of human binocular vision, refers to the dominant eye being more sensitive to detail discrimination than the non-dominant eye [19], [36]. Shneor *et al.* [20] observed that the dominant eye takes precedence in visual processing and can suppress input from the non-dominant eye when viewing targets of varying sizes under non-rivalry conditions. Similarly, Koçtekin *et al.* [21] reported that the dominant eye outperforms the non-dominant eye in red/green color discrimination. Oishi *et al.* [37] revealed that the dominant eye shows enhanced performance in horizontal saccades during reading tasks.

Meng *et al.* [23] applied the Eye dominance model to the Foveated Rendering (EFR) technique. Using the KFR method, they rendered the dominant eye's scene at a standard foveation level and the non-dominant eye's scene at a higher foveation level, achieving improved acceleration compared to the original approach. However, as illustrated in Fig. 1, Meng *et al.* employed a single-parameter foveation across the entire visual field, resulting in a significant mismatch between the pixel sampling rate and the visual acuity fall-off model in peripheral regions. This mismatch increased the number of rendered pixels, leading to higher rendering costs. Although KFR can better approximate the visual acuity fall-off model by controlling the pixel spatial distribution parameter α to 1, experiments indicate that this reduction degrades visual perception quality [13]. To address this issue, EFR experiments fixed α at 4 and adjusted the foveation parameter σ to compress the rendering resolution.

III. METHODOLOGY

A. Overview

In this section, we introduce a dominant-eye-aware asymmetric foveated rendering method that jointly optimizes eccentricity and foveation parameter settings, which we refer to as DEA-FoR. The pipeline of DEA-FoR is illustrated in Fig. 2. In contrast to our previous work [24], DEA-FoR enables users to customize eccentricity layer sizes based on the traits of their eyes and supports asymmetric configurations between the dominant and non-dominant eyes. DEA-FoR employs single-parameter foveation to adjust the foveation level of each layer, while multi-parameter foveation governs the overall foveation across the entire image for each eye. This asymmetric design allows for more efficient allocation of rendering resources and explores the differences in visual acuity between the dominant and non-dominant eyes.

The remainder of this section is structured as follows: 1) We describe the fundamental foveated rendering algorithm employed in our method in subsection III-B. 2) We present a generalized multi-parameter foveation approach in subsection III-C. 3) We introduce the incorporation of asymmetric eccentricity-based layer partitioning into our rendering pipeline in subsection III-D. 4) We describe the system implementation of our DEA-FoR in subsection III-E.

B. Basic Foveated Rendering Technique

Our objective is to utilize foveated rendering to accelerate the deferred shading pipeline. To this end, our framework is built on a state-of-the-art foveated rendering technique that integrates spatial factors into the rendering process. In recent years, several foveated rendering methods based on multi-spatial resolution have been proposed, including KFR [13] and RMFR [14]. Meng *et al.* extended KFR by incorporating eye-dominance guidance into their foveated rendering approach [23]. Our pilot test reveals that RMFR produces finer image details while minimizing rendering time and artifacts. Consequently, we adopt RMFR as the foundation of our framework.

The RMFR model operates in two passes. In the first pass, the textures in the G-buffer are downsampled using a rectangular mapping transformation. In the second pass, the transformed G-buffer performs a lighting pass and is rendered onto the full-resolution screen through inverse rectangular mapping.

Let the resolution of the full-resolution G-buffer be $W \times H$ and that of the transformed G-buffer be $w \times h$. RMFR controls the level of foveation using a parameter σ . This parameter is defined as the ratio between the width (or height) of the full-resolution G-buffer and the width (or height) of the transformed G-buffer. Formally, this is expressed as:

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

In the G-buffer space, the foveal point (x_f, y_f) is used as a reference to map each pixel coordinate (x, y) to a relative coordinate (x', y') through the following transformation:

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

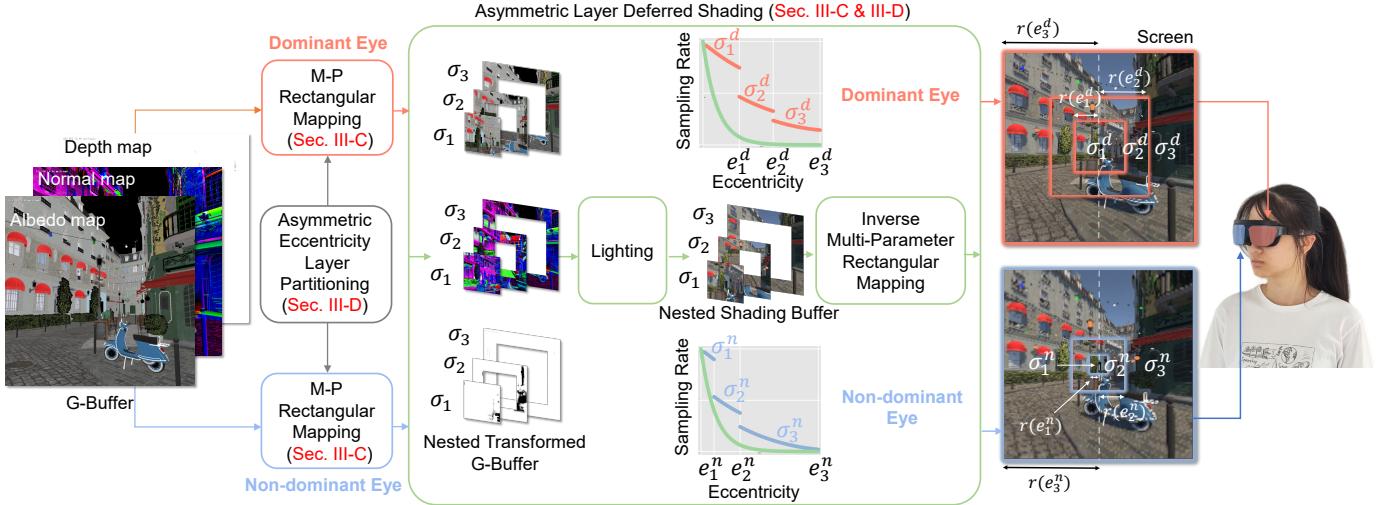


Fig. 2: The pipeline of DEA-FoR is seamlessly integrated with the deferred shading pipeline. Multi-Parameter (M-P) rectangular mapping is used to process the G-buffer. In this step, eccentricity layer sizes are customized based on dominant eye effect, enabling asymmetric configurations. During the lighting pass, pixels are shaded using data from the transformed G-buffer. Finally, the inverse M-P rectangular mapping is applied to the nested shading buffer, reconstructing the full-resolution screen.

The RMFR then maps the point (x', y') to (u, v) in the transformed G-buffer space using Equation 3. For simplicity, we take the case where $x' > 0$ and $y' > 0$:

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

where f_x and f_y are tunable parameters that control the sampling distributions along the horizontal and vertical directions, as defined by RMFR.

In the second pass, the mapped point (u, v) is transformed back into the G-buffer space, specifically to the full-resolution screen space coordinates (x'', y'') , using the following inverse rectangular mapping:

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

In summary, σ regulates the resolution of the transformed G-buffer to accelerate rendering, while f_x and f_y control the sampling distribution along horizontal and vertical directions. Adjusting these parameters can approximate the visual acuity fall-off model. For instance, decreasing f_x increases sampling density in the central region while reducing it in the periphery. However, Ye *et al.* [14] reported that even with optimal f_x and f_y , σ can only be increased up to 2.6, resulting in limited acceleration. In contrast, our experiments [24] show that, using the same f_x and f_y values, the σ in the central region of the dominant eye aligns with Ye *et al.*'s result ($\sigma = 2.51$). In the peripheral region beyond 22.5 degrees from the center, σ can be increased significantly, reaching an average value of 4.8. This adjustment leads to notable improvements in rendering acceleration. We conclude that using multiple σ to modulate the foveation across the entire image provides more noticeable benefits in rendering speed-up. Therefore, in our experiments,

we use $f_x = 0.38W$ and $f_y = 0.38H$ as suggested by [14], adjusting the foveation level solely through σ .

C. Multi-Parameter Optimization of the Rendering Pipeline

Based on the above rendering framework, we optimize it using multi-parameter foveation. This subsection focuses on optimizing the rendering for a single eye, similar to the process illustrated in Fig. 2, within the context of a deferred shading pipeline. The pipeline's input G-Buffer, which is the output of the rasterization process, consists of the image where each pixel contains color and depth information. DEA-FoR applies multi-parameter rendering pipeline to the image for each eye independently. Given that the lighting process is the most time-consuming stage in image rendering, DEA-FoR aims to reduce the number of pixels requiring shading during this process. In the following section, we introduce the multi-parameter rectangular mapping and its inverse process.

Multi-Parameter (M-P) Rectangular Mapping of the G-Buffer. For each eye, the G-buffer is partitioned into n concentric layers based on the eccentric angle relative to the fixation point. The n layers are denoted by $L_1, L_2, \dots, L_{n-1}, L_n$. We denote the tunable parameters of the layers as (\mathbf{E}, Σ) , where $\mathbf{E} = (e_1, e_2, \dots, e_{n-1}, e^*)$ and $\Sigma = (\sigma_1, \sigma_2, \dots, \sigma_{n-1}, \sigma_n)$. Here, \mathbf{E} represents the angular range of each layer, defined as $[e_0, e_1], [e_1, e_2], \dots, [e_{n-1}, e^*]$, with the constraint $0 = e_0 < e_1 < e_2 < \dots < e_{n-1} < e^*$. Additionally, e^* denotes the maximum eccentric angle of the visual field. Σ represents the multi-parameter foveation, where σ is defined in Equation (1).

After applying the M-P rectangular mapping to the n layers, we obtain n transformed G-buffers, where each corresponds to a layer of the G-buffer. The i -th transformed G-buffer is determined by an eccentricity range $[e_{i-1}, e_i]$ and a foveation parameter σ_i , where $i = 1, 2, \dots, n$. For each pixel at coordinate (x, y) in the original G-buffer, its relative position (x', y') with respect to the foveal point is computed using Equation (2). If

the point (x', y') lies within the i -th layer L_i , the mapping is performed using Equation (3) with the parameter σ_i , producing the i -th transformed G-buffer.

We compute the size of the i -th transformed G-buffer under the assumption that the user's gaze is centered at $(W/2, H/2)$ to simplify the analysis. The eccentric angle e_i ($0 \leq e_i \leq e^*$) denotes the angle between the gaze direction and the ray extending from the eye to the right edge of the layer along the horizontal plane. The local coordinates, as projected onto the window plane, are defined as:

$$P_{\text{local}} = (\tan e_i, 0, 1)^T. \quad (5)$$

By applying the Model-View-Projection (MVP) transformation followed by screen projection [3], the 2D screen coordinates corresponding to the eccentric angle e_i are obtained and denoted as P_{screen} :

$$P_{\text{screen}} = M_{\text{screen}} \cdot M_{\text{projection}} \cdot M_{\text{view}} \cdot M_{\text{model}} \cdot P_{\text{local}}. \quad (6)$$

The horizontal length of the i -th G-buffer, denoted as r_{xi} , is defined as the L2 distance from P_{screen} to the viewport center $O = (0.5W, 0.5H)$:

$$r_{xi}(e_i) = \|OP_{\text{screen}}\|. \quad (7)$$

The $r_{xi}(e_i)$ is shown in Fig. 2. In this paper, the eccentric angle in the vertical direction is assumed equal to that in the horizontal direction, however, we allow it to be different. Thus, we also define r_{yi} , which is computed in a similar manner to r_{xi} . The viewport size of the i -th G-buffer is given by $2r_{xi} \cdot 2r_{yi}$. Therefore, the viewport size of the i -th transformed G-buffer can be expressed as $l_{xi} \cdot l_{yi}$, as described below:

$$\begin{cases} l_{xi}(e_i, \sigma_i) = \frac{2r_{xi}(f_x + \frac{W}{2})}{\sigma_i(r_{xi} + f_x)}, \\ l_{yi}(e_i, \sigma_i) = \frac{2r_{yi}(f_y + \frac{H}{2})}{\sigma_i(r_{yi} + f_y)}. \end{cases} \quad (8)$$

Unlike the approach proposed by Guenter *et al.* [12], our method further accelerates rendering by discarding pixels in overlapping regions. The effective viewport size for the i -th transformed G-buffer ($i \geq 1$) is given by:

$$V_i = l_{xi} \cdot l_{yi} - \frac{\sigma_{i-1}^2}{\sigma_i^2} \cdot l_{x(i-1)} \cdot l_{y(i-1)}. \quad (9)$$

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

Shading in the Nested Transformed G-Buffer. The lighting pass calculates the illumination for each pixel from various directions using information from the n transformed G-buffers. This step constitutes the majority of the rendering cost, as its computational expense is directly proportional to the number of pixels being shaded [18], [33]. Unlike the traditional deferred shading pipeline, which shades a fixed $W \times H$ pixels, DEA-FoR shades a variable number of pixels for each eye. The total number of shaded pixels is given as:

$$N(\mathbf{E}, \Sigma) = \sum_{i=1}^n V_i. \quad (10)$$

The theoretical speedup in the shading stage achieved by DEA-FoR for each eye can be expressed as:

$$S(\mathbf{E}, \Sigma) = \frac{W \cdot H}{N(\mathbf{E}, \Sigma)}. \quad (11)$$

Inverse M-P Rectangular Mapping to the Screen. After the lighting pass, the inverse mapping is applied to the nested shading buffer to render the final image onto the screen. The screen-space coordinate (x'', y'') for each pixel is computed using Equation (4) on the i -th shading buffer.

This mapping generates n layers F_1, \dots, F_n on the screen, with their sizes gradually increasing. These layers are assembled into a complete frame F using Equation (9), where L_0 is defined as \emptyset :

$$F(x, y) = F_i(x, y), \quad (x, y) \in L_i \text{ and } (x, y) \notin L_{i-1}. \quad (12)$$

The Number of Nested Layers. Ideally, the value of σ increases with eccentric angle as the G-buffer is divided into a greater number of nested layers. However, the actual rendering time does not significantly decrease as the number of layers increases. This is due to the additional overhead caused by the serial execution of rendering for multiple layers on the GPU pipeline. Moreover, increasing the number of layers requires additional time in user studies, potentially causing visual fatigue among participants. Guenter *et al.* proposed dividing the visual field into three eccentricity layers, with the size and sampling rate of each layer determined through a computation process based on parameters obtained from user studies [12]. Following this guideline, we adopt a three-nested-layer structure for each eye's visual field in our method.

D. Asymmetric Eccentricity Layer Partitioning

Building upon the above multi-parameter optimization for monocular imagery, DEA-FoR introduces an asymmetric eccentricity layer partitioning approach for the visual fields of both the dominant and non-dominant eyes. In [24], we predefined fixed eccentricity ranges for each layer, specifically $[0, 10]^\circ$, $[10, 22.5]^\circ$, and $[22.5, 45]^\circ$. However, this approach neglects individual differences and the varying perceptual sensitivities of each eye under different visual scenes.

To address this limitation, we propose an asymmetric eccentricity layer optimization technique. This method independently determines the optimal eccentricity ranges for each eye. For the dominant eye, the eccentricity set is denoted as $\mathbf{E}^d = \{e_1^d, e_2^d, e_3^d\}$, while for the non-dominant eye, it is represented by $\mathbf{E}^n = \{e_1^n, e_2^n, e_3^n\}$. In both cases, e_3^d and e_3^n represent the maximum eccentricity value, which is 45° for the VR HMD used in this study. The values of e_1^d and e_1^n may be identical or differ, depending on the user's visual perception. The same applies to e_2^d and e_2^n . Considering the foveation levels, we employ the parameter sets (\mathbf{E}^d, Σ^d) and (\mathbf{E}^n, Σ^n) to control the rendering of images for the dominant and non-dominant eyes, respectively. Sample parameter values are illustrated in the middle section of Fig. 2.

The shading differences between the dominant and non-dominant eyes arise from the varying sizes of the nested transformed G-buffers. Specifically, for the non-dominant eye, the rendered frame adopts higher levels of foveation, resulting in a smaller total area for the transformed G-buffers compared to the output for the dominant eye. However, this does not

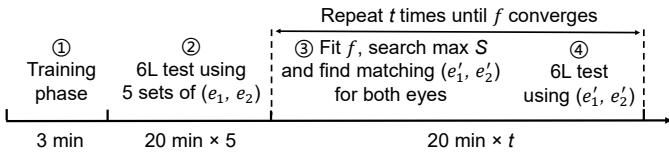


Fig. 3: The experiment procedure for preliminary study.

imply that individual layers for the non-dominant eye are necessarily smaller than those for the dominant eye, as discussed in subsection V-D (c). In summary, the fewer pixels requiring shading in the lighting pass substantially reduce computational overhead. The parameter sets (\mathbf{E}^d, Σ^d) and (\mathbf{E}^n, Σ^n) are derived from user studies described in Section V.

E. System Implementation

The DEA-FoR framework is designed to enhance rendering speed while preserving perceptual quality. The implementation details are outlined as follows:

Transition Between Layer Boundaries. DEA-FoR uses discrete pixel sampling rates at layer transitions. However, empirical user studies show that these transitions are imperceptible in the final rendered output. This effect can be explained by multiple factors: 1) Our tests found that boundary artifacts become perceptually significant only when the σ difference between consecutive layers exceeds 4. However, in our experiment, users select the foveation parameters while naturally maintaining visual quality, and we found the differences in σ between adjacent layers did not exceed 2.5 (dominant eye) and 3.5 (non-dominant eye). 2) The framework integrates Temporal Antialiasing (TAA) as a post-processing step following layer combination. This TAA approach not only mitigates temporal artifacts and aliasing in dynamic scenes but also effectively reduces potential boundary discontinuities. Additionally, Gaussian Blur is applied in peripheral regions [38], ensuring seamless transitional gradients.

The Rendering Engine. The DEA-FoR framework is implemented using the deferred shading pipeline of the Falcor engine [39], executed on an NVIDIA RTX 4090 laptop GPU. The Falcor engine provides precise timing metrics for each rendering step. In section V, the M-P rectangular mapping and deferred shading processes are grouped as Pass 1, while the inverse M-P rectangular mapping is categorized as Pass 2. For user studies, both rendering passes are implemented within the Unity Engine to simplify experimental procedures, following the methodology of Meng *et al.* [23].

IV. PRELIMINARY STUDY

While e_1 and e_2 ($e_1, e_2 \in \mathbf{E}$) theoretically take continuous values in the range $[0, 45]^\circ$, it is impractical to exhaust all possible combinations of these parameters in a user study. Therefore, in this study, we conduct a preliminary experiment to determine a rough range for e_1 and e_2 , which would facilitate more efficient and accurate eccentricity measurements in subsequent tests.

A. Apparatus and Participants

The user study takes place in a quiet indoor environment, using a desktop computer equipped with an NVIDIA GTX

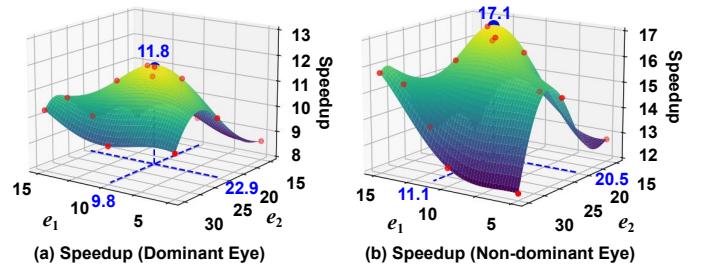


Fig. 4: The fitting results for P1 in the preliminary study. We use 2D gaussian function $S = G(e_1, e_2)$ to fit the measured speedup value and the eccentricity sets. The fitting process is repeated until $G(\cdot)$ converges.

1080 graphics card and an Intel Core i7-7700 CPU. The setup includes HUAWEI VR Glass, which provides a 90° field of view and a resolution of 3200×1600 (1600×1600 per eye). This device also supports myopic refraction within the range of $0\text{--}700^\circ$. For button-based interactions, the NOLO CV1 VR controller is used.

In this study, we use the *Fireplace Room* [40], as shown in Fig. 5 (a). We only collect data from three participants (three males, 22, 26, and 28 years old respectively) because each session takes three hours or more per participant in our study. Informed consent was obtained from all participants.

B. Procedure

The experimental procedure is illustrated in Fig. 3. We begin by randomly selecting an initial eccentricity pair (e_1, e_2) and allowing participants to familiarize themselves with it. Five sets of eccentricities (e_1, e_2) are then selected for both eyes during the experiment. These eccentricity sets are dispersedly distributed within the range $[0^\circ, 45^\circ]$ and are chosen as $(5^\circ, 22.5^\circ)$, $(10^\circ, 22.5^\circ)$, $(15^\circ, 22.5^\circ)$, $(10^\circ, 30^\circ)$, and $(15^\circ, 30^\circ)$. For each eccentricity set, we determine the optimal σ values for six layers of both eyes using the 6L method described in [24]. The speedup S for each (e_1, e_2) pair is then calculated using Equation (11). These results are subsequently used to fit a two-dimensional Gaussian function $G(\cdot)$ for each eye, providing the distribution of speedup:

$$S_d = G_d(e_1^d, e_2^d), S_n = G_n(e_1^n, e_2^n) \quad (13)$$

Using the $G(\cdot)$ data for each eye, we identify the maximum S and its corresponding (e'_1, e'_2) . The eccentricity set selected for the dominant eye is then used as the binocular parameter for further optimization. Next, the 6L test is conducted, and the collected data is employed to refine the fitting of $G(\cdot)$. This process is repeated t times until $G(\cdot)$ converges. The convergence condition is defined as follows, where k denotes the k -th iteration:

$$\max D - \min D \leq 3, D = \left\{ e'_1(k-i) \mid i = 0, 1, 2 \right\}, k \geq 5. \quad (14)$$

C. Results and Discussion

The optimal eccentricities measured for each participant's eyes are shown in Tab. I. The final data distribution and fitting results for the first participant are visualized in Fig. 4.

TABLE I: The optimal eccentricity sets for both eyes for three participants after iterations for many times.

Participant	Dominant Eye		Maximum S_d	Non-Dom. Eye		Maximum S_n	Sampling Iterations
	e_1^d	e_2^d		e_1^n	e_2^n		
1	9.8°	22.9°	11.8	11.1°	20.5°	17.2	13
2	10.3°	26.9°	7.9	9.8°	25.9°	19.6	8
3	5.0°	15.0°	24.4	8.7°	15.0°	25.0	8

Our experiments reveal that during the iterative process, the maximum S consistently corresponds to e_1 values within the range of [5, 15]° and e_2 values within [15, 30]°. Although this may be influenced by the initially chosen set of five eccentricity values, we believe that suitable eccentricity sets for both eyes of each participant can be identified within the ranges $e_1 \in [5, 15]$ ° and $e_2 \in [15, 30]$ °. Therefore, in our main experiments, e_1 values are selected from [5, 15]° and e_2 values from [15, 30]°.

Our preliminary experiment, conducted with a small number of participants, was used to determine the range of (e_1, e_2) values. For a larger number of participants, it is difficult to continuously vary e_1 or e_2 within the ranges of [5, 15]° or [15, 30]° to search for the maximum S . Therefore, we discretize these values within these intervals. To achieve this, we apply denser sampling near the center and sparser sampling towards the periphery. Specifically, we sample eight discrete sets for both the dominant and non-dominant eyes: (5°, 15°), (5°, 22.5°), (5°, 30°), (10°, 15°), (10°, 22.5°), (10°, 30°), (15°, 22.5°), (15°, 30°). In contrast, e_1 is set to 10° and e_2 to 22.5° in [24]. These discrete eccentricity sets will be used in subsequent experiments.

V. MAIN STUDY

The primary objective of this study is to explore whether asymmetric eccentricity layer partitioning for both eyes could further enhance the rendering acceleration compared to the previously fixed values of {10°, 22.5°}, while maintaining perceptual quality. In addition, we investigate how individual differences and scene texture complexity influence eccentricity partitioning and foveation for both eyes. We propose the following research questions:

RQ1: Does asymmetric eccentricity layer partitioning yield significant rendering acceleration?

RQ2: What are the individual user differences in binocular eccentricity and the maximum rendering acceleration?

RQ3: What are scene differences in binocular eccentricity and the maximum rendering acceleration?

A. Apparatus and Participants

We use the same apparatus as in the preliminary study. In this experiment, the Unity game engine is employed to render three scenes: *Fireplace Room* [40], *Amazon Lumberyard Bistro* [41], and *NVIDIA Emerald Square* [42], as shown in Fig. 5.

We recruit 22 participants (16 males and 6 females) from our laboratory and university campus, aged between 20 and 30 years (mean = 23.5, std = 2.2). All participants have normal or corrected-to-normal vision. Of these, 14 are right-eye dominant, while 8 are left-eye dominant. Informed consent was obtained from all participants.



Fig. 5: The following scenes are used in the user study: (a) *Fireplace Room* [40], (b) *Amazon Lumberyard Bistro* [41], (c) *NVIDIA Emerald Square* [42].

B. Data Collection Methodology

We aim to measure the level of foveation for both eyes in the configuration of 8 sets of $\{e_1, e_2\}$. For each $\{e_1, e_2\}$, in [24], we measure foveation parameters across 6 layers, taking about 20 minutes. Applying this method to all sets would require about 3 hours, causing visual fatigue and affecting measured results. To address this, we propose two hypotheses based on experimental observations to reduce redundant measurements, which shortens the testing time for 8 sets to 20 minutes in a single-scene test. Our hypotheses are:

H_1 : *For a given eye (e.g., the dominant eye), the foveation parameter σ_1 for the inner layer remains constant at maximum rendering speedup, regardless of e_1 value.*

H_2 : *For a given eye (e.g., the dominant eye), if e_1 is fixed, the foveation parameter σ_2 for the middle layer remains constant at maximum rendering speedup for any e_2 value.*

The core idea behind both hypotheses is that for any given layer, the perception of content closer to the fixation point determines the upper limit of the foveation parameter for that layer. For a detailed explanation of the two hypotheses mentioned above, please refer to the supplementary material. Based on H_2 , we further argue that if e_2 is fixed, the σ_3 in $[e_2, e_3]$ becomes determined. Therefore, we propose that the three-nested layer structure can be constructed through specific combinations of two-nested layers, as illustrated in Fig. 7.

The specific design is as follows. We use two nested layers, with e_1 as the only controllable eccentricity. Based on H_1 , the σ_1 of the inner layer only needs to be measured once. We sample e_1 from {5°, 10°, 15°, 22.5°, 30°} and measure σ_2 in different configurations. When constructing the three-nested layer structure, for example, if e_1 is 10° and e_2 is 22.5°, we use σ_1 as the foveation for the inner layer, $\sigma_2(10)$ as the foveation for the middle layer, and $\sigma_2(22.5)$ as the foveation for the outer layer, i.e., σ_3 .

This approach requires only six measurements per eye. We first measure the eccentricity and foveation parameters for the dominant eye, then fix these parameters to measure the non-dominant eye. The parameters for each eye are selected as the optimal sets for both eyes. To validate the rationality of our method, we conduct a comparative experiment with [24].

C. Procedure

To address the previously mentioned research questions and validate the rationality, we conduct two experiments: the Main Test and the Validation Test.

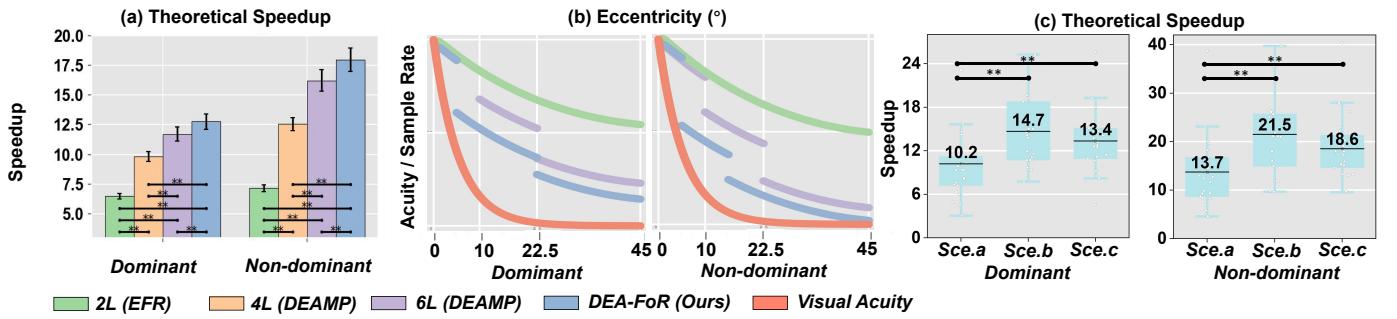


Fig. 6: Bar and line charts illustrating the performance of foveated rendering methods under different measurements: (a) Comparison of theoretical speedup, (b) Variation in sampling rate and visual acuity fall-off model, (c) Theoretical speedup of our method across three scenes.

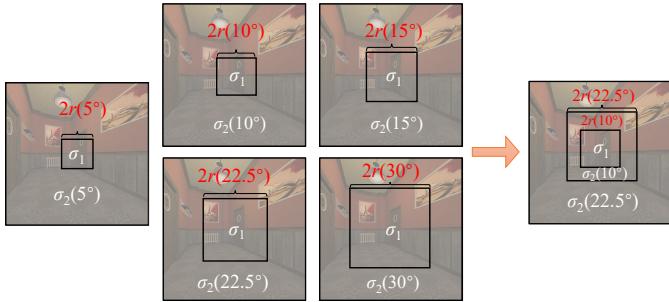


Fig. 7: The measurement process for foveation parameters in two-nested layer images and the construction process for three-nested layer images.

1) *Main Test*: The objective of the Main Test is to determine an optimal set of parameters that maximize rendering acceleration while preserving perceptual quality. Participants first identify their dominant eye using the Miles Test. Regarding the measurement of eccentricity and foveation parameters for the dominant and non-dominant eyes, we provide extensive details in Section B of the supplementary material and strongly recommend readers review that section. Overall, we believe that binocular visual perception quality is dominated by the dominant eye, and the lower bound of the maximum rendering acceleration for both eyes is determined by the dominant eye. Therefore, measurements of parameters are conducted sequentially, starting with the dominant eye and then followed by the non-dominant eye. When measuring the dominant eye, both eyes use identical rendering parameters. When measuring the non-dominant eye's parameters, the dominant eye maintains the previously determined optimal parameters unchanged, whereas only the non-dominant eye's rendering settings are adjusted. Using these measured σ parameters from two-nested layer structures, three-nested layer images are constructed as shown in Fig. 7.

2) *Validation Test*: To assess the validity of our approach, the Validation Test evaluates whether the measured speedups for the three-nested layer images are statistically consistent with the maximum speedup obtained in the Main Test.

This experiment employs the 6L Test described in [24]. Unlike the Main Test, the σ_3 value is determined based on the perceptual quality of the entire three-nested layer image. During the experiment, the eccentricity settings for both eyes

are configured using the $(e_1^d, e_2^d, e_1^n, e_2^n)$ values obtained in the Main Test, which correspond to the maximum calculated speedup. After completing the measurement, we calculate the rendering speedup.

D. Result

In this section, we present the results of our user study, analyzed primarily using RM-ANOVAs ($\alpha = 0.05$) and pairwise t-tests. Unless otherwise specified, all data reported meet the prerequisites for these statistical analyses. The results are presented from the following four aspects.

a) *Rendering acceleration comparison among different methods*: Regarding RQ1, We compare the performance of all methods in Fig. 6(a). RM-ANOVA showed that the four methods were significantly different on **theoretical speedup** ($F_{3,63} = 50.157, p < 0.001$ for the dominant eye; $F_{3,63} = 75.935, p < 0.001$ for the non-dominant eye). Pairwise t-tests confirmed that our method achieved a higher speedup than the other three methods for the dominant eye (all $p < 0.001$), with similar results for the non-dominant eye. We also implemented the deferred shading pipeline for all methods, and the **rendering times** are summarized in Tab. II. For columns 3–5, we used the foveation parameters σ from previous work [24], while for columns 6–8, we used the average foveation parameters measured in this study. For our method, we selected asymmetric eccentricity values and adjusted the foveation parameters to ensure the average theoretical speedup matched the results in Fig. 6(a). Specifically, DEA-FoR achieved a speedup of $4.2 \times$ compared to full-resolution rendering, while DEAMP achieved $4.0 \times$, and EFR achieved $3.6 \times$. A comparison of these methods is also provided in Tab. II.

We evaluated the efficacy of the approximation to the visual acuity model. Fig. 6(b) compares the visual acuity fall-off model with the normalized sampling rates of 2L, 6L, and our DEA-FoR. The average foveation parameter from the user study was used as the σ value for 2L and 6L, while for DEA-FoR, we selected foveation parameters that matched the average theoretical speedup. The 2L method showed a significant mismatch between the pixel sampling rate and the visual acuity fall-off model, leaving room for optimization in peripheral regions. In contrast, 6L approximated the model in a piecewise fashion. Our DEA-FoR aligned more closely with the visual acuity fall-off model than 6L. Additionally, the

TABLE II: A comparison of rendering time and frame rates (in FPS) is conducted among full-resolution rendering, KFR [13], RMFR [14], EFR [23], 2L (DEAMP) [24], 4L (DEAMP) [24], 6L (DEAMP) [24], and DEA-FoR (Ours) on binocular screens with a resolution of 1600×1600 per eye.

Procedure (ms)	Full-resolution	KFR [13]	RMFR [14]	2L (EFR) [23] dom n-dom	2L (DEAMP) [24] dom n-dom	4L (DEAMP) [24] dom n-dom	6L (DEAMP) [24] dom n-dom	DEA-FoR (Ours) dom n-dom
Depth Pass	0.11	0.11	0.11	0.11 0.11	0.11 0.11	0.11 0.11	0.11 0.11	0.11 0.11
Shadow Pass	0.16	0.15	0.15	0.15 0.15	0.15 0.15	0.15 0.14	0.14 0.14	0.14 0.15
Defer Pass	3.41	3.15	3.08	3.15 2.97	3.09 3.07	2.99 2.94	2.94 2.93	2.89 2.88
Skybox	0.01	0.01	0.01	0.01 0.01	0.01 0.01	0.01 0.01	0.01 0.01	0.01 0.01
Shading/Pass1	20.56	5.06	3.02	4.11 1.98	3.04 2.73	2.79 2.45	2.82 2.38	2.55 2.22
Pass2	/	0.13	0.09	0.13 0.08	0.09 0.09	0.06 0.06	0.06 0.05	0.06 0.05
TAA	0.27	0.26	0.24	0.25 0.24	0.24 0.24	0.23 0.23	0.23 0.23	0.23 0.23
Total	24.52	8.87	6.7	7.91 5.54	6.73 6.4	6.34 5.94	6.31 5.85	5.99 5.65
Binocular Time (ms)	49.04	17.74	13.40	13.45	13.13	12.28	12.16	11.64
Fps	20.4	56.4	74.6	74.3	76.2	81.4	82.2	85.9

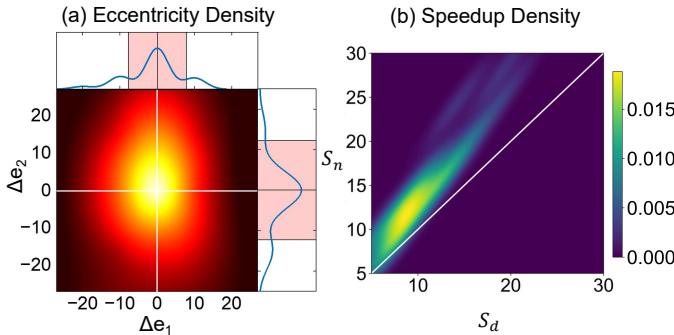


Fig. 8: (a) Probability density of eccentricity differences between both eyes. The horizontal axis represents $\Delta e_1 = (e_1^d - e_1^n)$. The vertical axis represents $\Delta e_2 = (e_2^d - e_2^n)$. Lighter colors indicate higher probabilities. White contours show marginal distributions. Pink areas denote $(\mu \pm \sigma)$ (std). (b) Probability density of the theoretical speedup for both eyes. The horizontal axis is S_d and the vertical axis is S_n .

non-dominant eye was observed to align better with the model than the dominant eye. In summary, our DEA-FoR is highly effective for visual acuity modeling.

b) *Binocular perception differences among individuals:* To address **RQ2**, we visualized differences in optimal eccentricity sets for binocular vision considering all scenes, as shown in Fig. 8(a). Differences between the dominant and non-dominant eyes are defined as $(\Delta e_1, \Delta e_2)$, where $\Delta e_1 = (e_1^d - e_1^n)$ and $\Delta e_2 = (e_2^d - e_2^n)$. This allows a direct comparison of the eccentricity settings between the two eyes. Key observations are as follows: 1) The data show more variation in the Δe_2 direction compared to Δe_1 , indicating that differences in e_2 between the two eyes vary more significantly across users. 2) The distribution is not concentrated at the origin, showing that the optimal eccentricity sets for the dominant and non-dominant eyes differ. To validate this, we calculated $\Delta E_1 = |\Delta e_1| + |\Delta e_2|$ and conducted a paired t-test comparing ΔE_1 with 0. Results show $t_{65} = 8.64$, $p < 0.001$, confirming the difference. 3) The data are roughly symmetric around the origin, with points distributed across all four quadrants without bias, suggesting individual variability in binocular eccentricity sets. 4) The marginal probability distributions (pink-shaded areas) are relatively uniform, with positive and

TABLE III: Statistical characteristics of eccentricity layer sizes in the three scenes. For each item, the same font of two layer sizes means that the difference between the layer sizes is not statistically significant (pairwise t-tests, $p > 0.05$) while different fonts indicate statistical significance. Layer sizes with an orange background indicate significant differences between them.

Eccentric. layer	Scene a	Scene b	Scene c
e_1^d	10.0°	10.2°	9.3°
e_2^d	<u>23.5°</u>	26.9°	26.6°
e_1^n	10.7°	10.5°	10.9°
e_2^n	25.2°	24.5°	25.2°

negative values evenly distributed. This indicates no overall size difference in eccentricities between the dominant and non-dominant eyes across all scenes. Furthermore, we computed $\Delta E_2 = \Delta e_1 + \Delta e_2$ and performed a paired t-test comparing ΔE_2 with 0. The results show $t_{65} = -0.147$, $p = 0.883$, supporting this conclusion.

Next, we plotted Fig. 8(b) to examine the relationship between the theoretical speedup of the dominant and non-dominant eyes. Each point in the figure represents a speedup pair (S_d, S_n) for the two eyes. The following observations were made: 1) Speedup varies notably across individuals. The dominant eye's speedup mostly ranges from 5 to 15, while the non-dominant eye's spans 5 to 20, showing greater variability. 2) Most points lie above the diagonal line, indicating the non-dominant eye generally has a higher speedup than the dominant eye. 3) A positive correlation exists: as the dominant eye's speedup increases, so does the non-dominant eye's.

c) *Binocular perception differences among different scenes:* We plotted Fig. 6(c) to show the distribution of optimal binocular acceleration across scenarios. Results reveal significant differences in acceleration between scenarios ($F_{2,42} = 5.01$, $p = 0.01$ for the dominant eye; $F_{2,42} = 6.19$, $p = 0.004$ for the non-dominant eye). Post-hoc t-tests found that, for the dominant eye, acceleration in Scene (a) was significantly lower than in Scenes (b) and (c) ($p = 0.003$ and $p = 0.034$), with no difference between Scenes (b) and (c). Similarly, for the non-dominant eye, acceleration in Scene (a) was significantly lower than in Scenes (b) and (c) ($p = 0.001$ and $p = 0.034$),

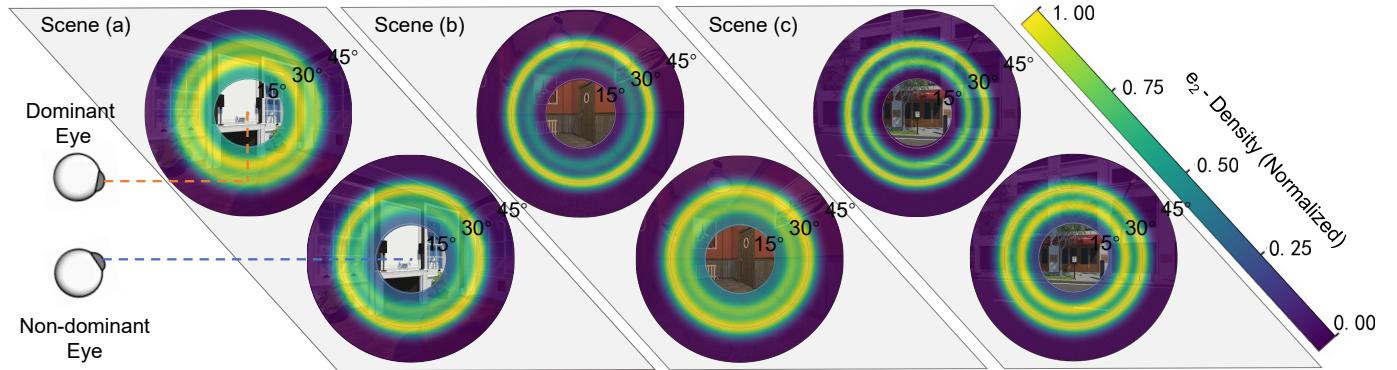


Fig. 9: The probability density of e_2 for the two eyes across three scenes. The probability density at each eccentricity is normalized. The eccentricities of 15° and 30° are marked with two circles.

with no difference between Scenes (b) and (c).

To explore variations in optimal eccentricity layers across scenarios, we calculated Table III and plotted Fig. 9. Table III shows the mean optimal eccentricity for each eye, while Fig. 9 visualizes the e_2 distribution across scenarios for all users. Row-wise significance tests in Table III showed significant differences in e_2^d ($F_{2,42} = 3.425, p = 0.039$). Post-hoc t-tests revealed e_2^d in Scene (a) was significantly smaller than in Scenes (b) and (c) ($p = 0.021$ and $p = 0.036$). Similarly, the first row of Fig. 9 shows that e_2^d in Scene (a) is more centered near the fixation point than in the other scenarios.

Column-wise t-tests in Table III showed that, for Scene (b), e_2^d was significantly larger than e_2^n ($p = 0.05$). Fig. 9 (second column) supports this, showing that the non-dominant eye's e_2 distribution is slightly more centered than the dominant eye's. No other significant differences in optimal eccentricity were observed. See Section VI for further discussion.

d) Validation test results: We conducted a validation test to assess the rationality of our data collection method. The results are presented in Fig. 1 of the Supplemental Material. They indicate no significant differences in speedup between the two methods (dominant eye: $t_{21} = 0.546, p = 0.591$; non-dominant eye: $t_{21} = 1.031, p = 0.314$). This confirms that our approach of constructing three-nested layers using two-nested layers is equivalent to direct measurement.

VI. DISCUSSION

Our main study validated that the asymmetric eccentricity layer partitioning for both eyes yields significant acceleration. Additionally, individual differences and scene texture complexity were found to influence eccentricity partitioning and foveation in both eyes. We discuss these results in detail below.

1) Significant Differences in Eccentricities and Acceleration Across Users: Fig. 8(a) shows that optimal binocular eccentricity sets vary significantly among individuals. The differences in e_2 values between the two eyes are more pronounced than those in e_1 . This may be because, although the non-dominant eye is less sensitive to visual stimuli overall, its central region still maintains relatively high visual acuity. Consequently, e_1 remains relatively consistent across users. In contrast, the peripheral regions exhibit greater individual vari-

ability, leading to more substantial differences in e_2 between the eyes.

Furthermore, a positive correlation is observed between the acceleration of the dominant and non-dominant eyes across users. However, users with higher overall acceleration tend to show larger differences in acceleration between their two eyes. The non-dominant eye also exhibits greater variability in acceleration across individuals, suggesting significant individual differences in the dominant-eye effect. For users with lower overall visual sensitivity, the dominant-eye effect is typically stronger, which explains Fig. 8(b) where the difference ($S_n - S_d$) increases rapidly as S_d increases. Applying distinct rendering parameters for the dominant and non-dominant eyes improves acceleration performance for these users.

2) Significant Differences in Eccentricities and Acceleration Across Scenes: The optimal acceleration for Scene (a) is significantly lower than that of the other two scenes, which we attribute to the higher visual complexity of Scene (a). Compared to the other scenes, Scene (a) features more intricate textures and colors, making artifacts at the edges of the rendered images more noticeable to users. This leads to reduced foveation levels and lower overall acceleration.

As shown in Fig. 9 and Table III, the e_2 value for the dominant eye in Scene (a) is significantly smaller than in the other two scenes, reflecting the impact of its complex textures. In contrast, Scenes (b) and (c) contain large areas of uniform color and texture beyond the 15° region. These details in Scene (a) increase user sensitivity to peripheral edge artifacts, resulting in a relatively lower σ value. In this case, the values of $\sigma_2(15^\circ)$, $\sigma_2(22.5^\circ)$, and $\sigma_2(30^\circ)$ are closer (ranging from 3.1 to 4.2). As σ gradually increases from the innermost to the outermost layer, selecting a smaller eccentricity (e.g., 15° or 22.5°) helps maximize acceleration.

For the non-dominant eye, however, the e_2 value for Scene (a) is not smaller than those for the other two scenes. This is likely due to the reduced sensitivity of the non-dominant eye in the peripheral region. In Scene (a), the values of $\sigma_2(15^\circ)$, $\sigma_2(22.5^\circ)$, and $\sigma_2(30^\circ)$ vary more significantly (ranging from 3.7 to 5.8). As a result, the non-dominant eye tends to select a larger eccentricity to maximize acceleration.

VII. KEY FINDINGS AND DESIGN RECOMMENDATIONS

Based on the results and analysis presented above, we offer the following actionable implications for future research as follows:

1) Our dominant-eye-aware asymmetric foveated rendering method (DEA-FoR) outperforms other foveated rendering techniques in terms of rendering speed. It significantly reduces the number of pixels shaded during the lighting pass.

2) Differences in e_2 values between the two eyes are more pronounced than those in e_1 . Stronger dominant-eye effects are observed in users with lower overall visual sensitivity. Applying greater foveation to the peripheral regions of the non-dominant eye's image significantly improves acceleration performance, particularly for these users.

3) Experimental results indicate that eccentricity layer sizes are influenced by scene texture complexity. In scenes with simple textures, the outermost layer eccentricity of the dominant eye is larger, while that of the non-dominant eye is smaller. In contrast, in scenes with complex textures, the outermost layer eccentricity of the dominant eye is smaller, while that of the non-dominant eye is larger.

4) To support practical applications and scene development, we propose a design recommendation based on the minimum acceptable foveation level for 75% of users in both simple and complex scenes. For complex scenes (e.g., Fig. 5 (a)), the recommended parameters for the dominant eye are $(e_1^d, e_2^d) = (10.0^\circ, 23.5^\circ)$ and $(\sigma_1^d, \sigma_2^d, \sigma_3^d) = (2.0, 2.5, 3.1)$, while for the non-dominant eye, they are $(e_1^n, e_2^n) = (10.7^\circ, 25.2^\circ)$ and $(\sigma_1^n, \sigma_2^n, \sigma_3^n) = (2.5, 2.7, 3.4)$. For simple scenes (e.g., Fig. 5 (b) and (c)), the recommended parameters for the dominant eye are $(e_1^d, e_2^d) = (10.2^\circ, 26.9^\circ)$ and $(\sigma_1^d, \sigma_2^d, \sigma_3^d) = (2.3, 2.9, 4.5)$, while for the non-dominant eye, they are $(e_1^n, e_2^n) = (10.5^\circ, 24.5^\circ)$ and $(\sigma_1^n, \sigma_2^n, \sigma_3^n) = (2.8, 3.3, 5.1)$.

VIII. LIMITATIONS AND FUTURE WORK

The exploration of optimal eccentricity sets (e_1, e_2) across different scenes is currently restricted to sampling eight discrete sets within the ranges $[5^\circ, 15^\circ]$ and $[15^\circ, 30^\circ]$. However, the true optimal eccentricity may fall outside these ranges or not align with the sampled discrete values within them. To address these limitations, future research should focus on developing more efficient methods to explore larger intervals and evaluate continuous samples within these ranges, while accounting for experimental duration constraints.

The theoretical speedup is inversely proportional to the number of pixels rendered. However, the actual rendering time deviates from this prediction due to challenges in parallelizing the rendering of multiple nested layers within the GPU pipeline. These challenges can largely be attributed to limitations in the Falcor framework. Future research will focus on developing parallel rendering techniques for nested layers to enhance rendering efficiency.

For the practical application of our method, the eccentricity and foveation parameters recommended in Section VII can be applied to both simple and complex scenes. Alternatively, users may retest and determine optimal parameters themselves.

This process takes approximately 20 minutes in our experiments. In future work, we aim to develop an algorithm for the efficient selection of these parameters. For example, inspired by Krajancich's work [43], we plan to explore real-time adaptive eccentricity sets based on the distribution of spatial visual attention. Besides, automated dominant eye detection methods could also be explored [44], which would enable the integration of our proposed method into practical VR applications.

IX. CONCLUSION

In this paper, we proposed DEA-FoR, a Dominant-Eye-aware Asymmetric Foveated Rendering method that improves rendering efficiency while preserving perceptual quality. The method retains the three-layer structure for each eye's visual field. It introduces eccentricity as a new variable, allowing users to customize eccentricity sets and enabling asymmetric configurations between the dominant and non-dominant eyes. An efficient parameter selection approach was also introduced, reducing experiment duration and enhancing practicality. User studies validated DEA-FoR's effectiveness and revealed the influence of individual differences and scene texture complexity on eccentricity and foveation settings. This work not only improves rendering performance but also provides valuable insights into binocular vision perception.

ACKNOWLEDGMENTS

This work was supported by Beijing Natural Science Foundation (L242019), the China Postdoctoral Science Foundation (2025M774238) and the Postdoctoral Fellowship Program of CPSF (GZC20252741).

REFERENCES

- [1] Z. Wang, Y. Zhao, and F. Lu, "Gaze-Vergence-Controlled See-Through Vision in Augmented Reality," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 11, pp. 3843–3853, 2022.
- [2] 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)*, 2023, pp. 22–32.
- [3] 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*, vol. 51, no. 5, pp. 524–534, 2021.
- [4] 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)*. IEEE, 2020, pp. 165–166.
- [5] 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*. IEEE, 2022, pp. 548–549.
- [6] L. Wang, X. Shi, and Y. Liu, "Foveated rendering: A state-of-the-art survey," *Computational Visual Media*, vol. 9, no. 2, pp. 195–228, 2023.
- [7] Z. Wang, M. Rao, S. Ye, W. Song, and F. Lu, "Towards spatial computing: recent advances in multimodal natural interaction for xr headsets," 2025. [Online]. Available: <https://arxiv.org/abs/2502.07598>
- [8] S. Jabbireddy, X. Sun, X. Meng, and A. Varshney, "Foveated rendering: Motivation, taxonomy, and research directions," *arXiv preprint arXiv:2205.04529*, 2022.
- [9] 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.*, vol. 35, no. 6, dec 2016. [Online]. Available: <https://doi.org/10.1145/2980179.2980246>
- [10] T. Tariq, C. Tursun, and P. Didyk, "Noise-based enhancement for foveated rendering," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 4, pp. 1–14, 2022.

- [11] X. Meng, R. Du, J. F. JaJa, and A. Varshney, “3D-kernel foveated rendering for light fields,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 8, pp. 3350–3360, 2020.
- [12] B. Guenter, M. Finch, S. Drucker, D. Tan, and J. Snyder, “Foveated 3D Graphics,” *ACM Trans. Graph.*, vol. 31, no. 6, nov 2012. [Online]. Available: <https://doi.org/10.1145/2366145.2366183>
- [13] X. Meng, R. Du, M. Zwicker, and A. Varshney, “Kernel Foveated Rendering,” *Proc. ACM Comput. Graph. Interact. Tech.*, vol. 1, no. 1, jul 2018. [Online]. Available: <https://doi.org/10.1145/3203199>
- [14] 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)*, 2022, pp. 756–764.
- [15] L. Franke, L. Fink, J. Martschinke, K. Selgrad, and M. Stamminger, “Time-Warped Foveated Rendering for Virtual Reality Headsets,” *Computer Graphics Forum*, vol. 40, no. 1, pp. 110–123, 2021.
- [16] M. Koskela, T. Viitanen, P. Jääskeläinen, and J. Takala, “Foveated Path Tracing,” in *Advances in Visual Computing*. Cham: Springer International Publishing, 2016, pp. 723–732.
- [17] 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*, vol. 35, no. 7, pp. 289–298, 2016. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13026>
- [18] 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.*, vol. 38, no. 4, jul 2019. [Online]. Available: <https://doi.org/10.1145/3306346.3322985>
- [19] C. Porac and S. Coren, “The dominant eye.” *Psychological bulletin*, vol. 83, no. 5, p. 880, 1976.
- [20] E. Shneor and S. Hochstein, “Eye dominance effects in conjunction search,” *Vision Research*, vol. 48, no. 15, pp. 1592–1602, 2008. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0042698908002344>
- [21] B. Koçtekin, N. Ünal 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*, vol. 6, no. 15, pp. 733–8, 2013 Oct.
- [22] B. Chaurasia and B. Mathur, “Eyedness,” *Acta Anatomica*, vol. 96, no. 2, pp. 301–305, 07 2008. [Online]. Available: <https://doi.org/10.1159/000144681>
- [23] X. Meng, R. Du, and A. Varshney, “Eye-dominance-guided Foveated Rendering,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 5, pp. 1972–1980, 2020.
- [24] Z. Wang, X. Gu, and F. Lu, “Deamp: Dominant-eye-aware foveated rendering with multi-parameter optimization,” in *2023 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2023, pp. 632–641.
- [25] R. Albert, A. Patney, D. Luebke, and J. Kim, “Latency Requirements for Foveated Rendering in Virtual Reality,” *ACM Trans. Appl. Percept.*, vol. 14, no. 4, sep 2017. [Online]. Available: <https://doi.org/10.1145/3127589>
- [26] 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)*, 2018, pp. 1–2.
- [27] J. Spjut, B. Boudaoud, J. Kim, T. Greer, R. Albert, M. Stengel, K. Aksit, and D. Luebke, “Toward standardized classification of foveated displays,” *IEEE transactions on visualization and computer graphics*, vol. 26, no. 5, pp. 2126–2134, 2020.
- [28] B. Krajancich, P. Kellnhofer, and G. Wetzstein, “A perceptual model for eccentricity-dependent spatio-temporal flicker fusion and its applications to foveated graphics,” *ACM transactions on graphics (TOG)*, vol. 40, no. 4, pp. 1–11, 2021.
- [29] R. Fan, X. Shi, K. Wang, Q. Ma, and L. Wang, “Scene-aware foveated rendering,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 30, no. 11, pp. 7097–7106, 2024.
- [30] M. Stengel, S. Grogorick, M. Eisemann, and M. Magnor, “Adaptive Image-Space Sampling for Gaze-Contingent Real-Time Rendering,” *Comput. Graph. Forum*, vol. 35, no. 4, p. 129–139, jul 2016.
- [31] I. Wald, P. Slusallek, C. Benthin, and M. Wagner, “Interactive rendering with coherent ray tracing,” in *Computer graphics forum*, vol. 20, no. 3. Wiley Online Library, 2001, pp. 153–165.
- [32] M. Fujita and T. Harada, “Foveated real-time ray tracing for virtual reality headset,” *Light Transport Entertainment Research*, 2014.
- [33] M. Koskela, A. Lotvonen, M. Mäkitalo, P. Kivi, T. Viitanen, and P. Jääskeläinen, “Foveated Real-Time Path Tracing in Visual-Polar Space,” 2019.
- [34] X. Shi, L. Wang, X. Liu, J. Wu, and Z. Shao, “Scene-aware foveated neural radiance fields,” *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–14, 2024.
- [35] 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*, vol. 28, no. 11, pp. 3854–3864, 2022.
- [36] F. Zhong, G. A. Koulieris, G. Drettakis, M. S. Banks, M. Chambe, F. Durand, and R. K. Mantiuk, “Dice: dichoptic contrast enhancement for vr and stereo displays,” *ACM Trans. Graph.*, vol. 38, no. 6, Nov. 2019. [Online]. Available: <https://doi.org/10.1145/3355089.3356552>
- [37] A. Oishi, S. Tobimatsu, K. Arakawa, T. Taniwaki, and J. ichi Kira, “Ocular dominance in conjugate eye movements at reading distance,” *Neurosci Res.*, vol. 52, no. 3, pp. 263–8, 2005 Jul.
- [38] “Gaussian blur,” [Online], https://en.wikipedia.org/wiki/Gaussian_blur, Accessed June 12, 2023.
- [39] “The Falcor rendering framework.” [Online], <https://github.com/NVIDIAAGameWorks/Falcor>, Accessed June 12, 2023.
- [40] M. McGuire, “Computer Graphics Archive,” July 2017, <https://casual-effects.com/data>. [Online]. Available: <https://casual-effects.com/data>
- [41] A. Lumberyard, “Amazon Lumberyard Bistro, Open Research Content Archive (ORCA),” July 2017, <http://developer.nvidia.com/orca/amazon-lumberyard-bistro>. [Online]. Available: <http://developer.nvidia.com/orca/amazon-lumberyard-bistro>
- [42] K. A. Nicholas Hull and N. Bentj, “Nvidia emerald square, open research content archive (orca),” July 2017, <http://developer.nvidia.com/orca/nvidia-emerald-square>. [Online]. Available: <http://developer.nvidia.com/orca/nvidia-emerald-square>
- [43] B. Krajancich, P. Kellnhofer, and G. Wetzstein, “Towards attention-aware foveated rendering,” *ACM Trans. Graph.*, vol. 42, no. 4, Jul. 2023. [Online]. Available: <https://doi.org/10.1145/3592406>
- [44] F. Prummer, F. Weidner, and H. Gellersen, “Advancing eye dominance testing: Comparing traditional methods with an in-hmd approach for ar/vr applications,” in *2025 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, 2025, pp. 112–116.

X. BIOGRAPHY SECTION

Zhimin Wang is a postdoctoral researcher with the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, China. He received his Ph.D. from Beihang University in 2024 and his B.S. from Chang'an University in 2019. His research focuses on VR/AR, human-computer interaction, and eye-tracking technologies. He serves as a program committee member for AAAI 2025 and as a regular reviewer for leading international venues, including IEEE VR, ISMAR, TVCG, CVPR, and IJHCI.



Xiangyuan Gu is currently a student with the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University. He obtained his bachelor's degree in computer science and technology from Beihang University, China, in 2024. His research focuses on virtual reality, eye-tracking and computer graphics.



Feng Lu (Senior Member, IEEE) received the BS and MS degrees in Automation from Tsinghua University, China in 2007 and 2010, respectively, and the PhD degree in Information Science and Technology from The University of Tokyo, Japan in 2013. He is currently a full Professor with the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University. His research interests include computer vision, natural interaction and VR/AR. He is a distinguished member of CCF and CSIG. He is serving/has served as an Area Chair for prestigious international conferences such as CVPR, ICCV, ECCV, NeurIPS and ACM MM.

