Gaze-Contingent Perceptual Level of Detail Prediction

L. Surace[1], C. Tursun[1,2], U. Çelikcan[1,3], P. Didyk[1]

[1]Università della Svizzera italiana, Switzerland
[2]University of Groningen, Netherlands
[3]Hacettepe University, Turkey

Figure 1: *The gaze-contingent perceptual level of detail allows reducing the number of primitives in a mesh in the peripheral vision without impacting the user experience based on viewing parameters. In this figure, we show two example meshes whose geometric complexity is reduced according to the predictions of our method as the meshes approach higher retinal eccentricities relative to the observer's gaze position located at the center.*

Abstract

New virtual reality headsets and wide field-of-view displays rely on foveated rendering techniques that lower the rendering quality for peripheral vision to increase performance without a perceptible quality loss. While the concept is simple, the practical realization of the foveated rendering systems and their full exploitation are still challenging. Existing techniques focus on modulating the spatial resolution of rendering or shading rate according to the characteristics of human perception. However, most rendering systems also have a significant cost related to geometry processing. In this work, we investigate the problem of mesh simplification, also known as the level of detail (LOD) technique, for foveated rendering. We aim to maximize the amount of LOD simplification while keeping the visibility of changes to the object geometry under a selected threshold. We first propose two perceptually inspired visibility models for mesh simplification suitable for gaze-contingent rendering. The first model focuses on spatial distortions in the object silhouette and body. The second model accounts for the temporal visibility of switching between two LODs. We calibrate the two models using data from perceptual experiments and derive a computational method that predicts a suitable LOD for rendering an object at a specific eccentricity without objectionable quality loss. We apply the technique to the foveated rendering of static and dynamic objects and demonstrate the benefits in a validation experiment. Using our perceptually-driven gaze-contingent LOD selection, we achieve up to 33% of extra speedup in rendering performance of complex-geometry scenes when combined with the most recent industrial solutions, i.e., Nanite from Unreal Engine.

CCS Concepts

• ***Computing methodologies*** → ***Perception;*** *Virtual reality;*

1. Introduction

Virtual reality devices offer exciting opportunities to create immersive 3D environments for a wide range of applications, from entertainment to education. However, they also pose significant challenges for rendering techniques, where high spatial and temporal resolutions are needed to provide sufficient visual quality and maintain a comfortable viewing experience. Despite constant progress in the efficiency and performance of graphics hardware, it is always computational limits that determine the viewer's visual experience. Therefore, it is critical, especially for near-eye wide-field-of-view displays, to save on computations whenever possible and focus on essential aspects of image generation for the viewer's experience. Foveated rendering techniques are critical to improving the efficiency of rendering on such devices. They exploit the limited sensitivity of the human visual system to spatial distortions in the periphery and lower the rendering quality accordingly without creating objectionable artifacts. Since rendering consists of geometry processing and shading stage, a foveation strategy can be applied to either of them. While many foveated rendering techniques focus on simplifying the shading stage by reducing the rendering resolution

or shading rate for peripheral vision, few works focus on gaining computational savings by reducing the mesh quality for peripheral vision (Figure 1). However, complex geometries are still challenging, which justifies the existence and development of techniques that minimize the number of rendered triangles, such as the Nanite virtualized geometry system in Unreal Engine [Epi23].

The rendering cost related to geometry is directly related to the number of primitives used to represent objects in the 3D scene. In many cases, meshes come from artistic pipelines or 3D scans, where the goal is to create ultimate representations of the geometry with a high amount of detail. The precision of the mesh often exceeds the quality requirements. A common example is when meshes in the scene are rendered at a further distance. The details contained in a high-resolution mesh cannot be rendered accurately due to the limitation in the spatial resolution of the display devices. A large class of techniques, so-called level-of-detail (LOD) [LRC*03], deals with this problem. The methods employ mesh simplification to compute lower-polygonal meshes, each specific to a given distance at which the object is rendered. During the rendering time, the methods try to minimize the overall number of rendered primitives by choosing appropriate versions of the meshes depending on the object's size on the screen, such that no visual artifacts are created. Similarly, it is possible to incorporate insights from perception when choosing an appropriate LOD for rendering. As the human sensitivity to distortions decreases towards the periphery, simpler meshes can be used to represent objects. The idea has been explored in previous works [OYT96; MD01], which propose determining the appropriate LOD for a particular eccentricity based on the size of the polygons within the mesh.

In this work, we introduce perceptually inspired visibility models for mesh simplification and use them in a new computational method that selects the simplest LOD possible while keeping the changes to object geometry below a selected visibility threshold. We propose spatial and temporal models to evaluate both aspects of visibility. The spatial model focuses on the visibility of changes in object silhouette and object body, whereas the temporal model aims at measuring the visibility of switching between different LODs in a dynamic scene. We calibrate both models on data from perceptual experiments that investigate human sensitivity to the geometry distortions of simplified meshes at different eccentricities. Finally, we demonstrate how we utilize the proposed models with our method to select the LOD according to the eccentricity at which the object is rendered in real time. For this, we test our method in a free-viewing application while rendering multiple static and dynamic objects and evaluate the performance.

2. Related work

The gaze-contingency paradigm aims to update the display contents according to an observer's gaze position. It has been studied in different fields, from marketing to user interface design [WP08; Jac95; Duc02; RLMS03]. Foveated rendering is a well-known gaze-contingent technique in computer graphics, which aims to couple the gaze position from an eye tracker with perceptual optimizations of the rendered content. There are many successful foveated rendering applications that adapt the rendering process according to nonuniform characteristics of the Human Visual System (HVS) across the visual field [DÇ07; PKS*16; MIGS22]. Earlier applications of foveated rendering aimed to provide the observer with the perception of high resolution over a wide field-of-view [BC88; Gle94]. More recently, Guenter et al. [GFD*12] reduced resolution with increasing eccentricity and used bilinear upsampling for foveated rendering. Swafford et al. [SIK*16] proposed practical rules for adjusting the rendering quality in foveated resolution, ambient occlusion, terrain tesselation, and ray casting. Furthermore, they proposed a new parameterization for the contrast sensitivity function in HDR-VDP-2 [MKRH11] for the peripheral visual field. Patney et al. [PSK*16] used foveated coarse-pixel shading and contrast enhancement as a post-processing step to improve perceived quality. Tursun et al. [TAW*19] introduced a foveated rendering method that adjusts the resolution as a function of both the retinal eccentricity and the underlying content. Tariq et al. [TTD22] took a step further in post-processing and synthesized additive noise to enhance perceived spatial detail in foveated rendering. Furthermore, some of the work in this area focused on improving image and video compression [TEHM96; KG96]. Kaplanyan et al. [KSL*19] recently introduced a foveated image reconstruction technique from sparsely sampled data based on deep learning.

In addition to these studies, which aim to reduce the image resolution or shading rate in the peripheral visual field, some work has been done on reducing the geometric complexity of a 3D model where the details cannot be perceived by an observer. Indeed, the geometries used in 3D scenes under standard rendering scenarios may often contain millions of primitives, such as triangles, that heavily impact the rendering time unless a mesh simplification method is employed.

In this context, there is a requirement for 2D and 3D shape similarity metrics to evaluate and tune simplification methods. For the 2D shape similarity, several metrics considered the shape contour or a particular region of interest [LL00; CLZ01; VL06]. Some studies also focused on representing shape contours using mathematical functions to measure similarity. For example, Kuhl and Giardina [KG82] represented the contour through elliptic functions, whereas Cortese and Dyre [CD96] used Fourier descriptors to measure the perceptual similarity of shapes. Loffler [Lof08] reviewed the perceptual mechanisms of the HVS for interpreting visual objects and reported evidence for both local and global processing. Another problem related to 2D shape similarity is the measurement of similarity in the image space. To this end, Park et al. [PLL06] introduced the perceptually modified Hausdorff distance. More recently, Mantiuk et al. [MDC*21] proposed the FovVideoVDP quality metric to evaluate similarity for both images and videos, depending on the retinal eccentricity of the stimulus.

For 3D shapes, the similarity problem was investigated by Shum et al. [SHI96]. Such type of metrics allowed to direct the research towards applications-oriented pipelines, e.g., Chen et al. [CDS*22] proposed a perceptually optimized 3D streaming method. Hasselgren et al. [HML*21] jointly optimized the triangle meshes and shading models to match the appearance of a reference scene.

Funkhouser and Séquin [FS93] introduced one of the earliest works on eccentricity-dependent LOD selection, which assumed a fixed gaze position at the center of the display. Hitchner and Mc-

Greevy [HM93] modeled the importance of objects as a function of several factors, including eccentricity. Reddy [Red96] developed a perceptual prediction method for reducing the polygonal complexity of an object. Ohshima et al. [OYT96]'s model degraded the LOD according to visual acuity, which is expressed as an exponential function of eccentricity. Tiwary et al. [TRK20] introduced an optimization method for adaptive geometric tessellation in foveated rendering. Murphy and Duchowski [MD01] introduced a degradation method based on eccentricity and applied it to geometries on a VR system.

Unlike existing eccentricity-dependent LOD selection methods, our method works in image space, which is a more accurate representation of what the observer sees. In addition to changes in the body of the object due to shading, a considerable amount of perceptually significant changes in an image are located near luminance and chrominance discontinuities that are also frequently observed around object silhouettes [AČMS10]. To maintain a good silhouette approximation, Mata and Pastor [MPR08] introduced a view-independent mesh simplification method. However, their method focuses only on silhouettes and is not perceptually calibrated for gaze-contingent applications, whereas we focus our analysis on changes in the object body as well as the object silhouette. Furthermore, we consider the effect of temporal changes resulting from switching between LODs in a dynamic scene. We develop an end-to-end system that can be used efficiently in modern rendering applications. We evaluate our proposed method with various rendering scenarios, which include objects with different silhouettes, materials, and a selected set of simple textures, whereas previous studies mostly used simple flat shading.

3. Overview

When applying LOD techniques, there are three main sources of visible distortions. First, poorly selected low-quality meshes can result in visible geometric distortions. Second, mesh inaccuracy can lead to visible spatial changes in shading due to inaccurate normal vectors or texture coordinates. Finally, dynamically changing the LOD of an object during rendering may become visible to an observer as so-called *popping artifacts*. To address these three sources of visible distortions, we propose a spatial model (Section 4.1) and a temporal model (Section 4.2). We calibrate both models with a new perceptual dataset for LOD distortions (Section 5). Next, we introduce a gaze-contingent LOD selection method (Section 6), which exploits the predictions of the proposed models in a real-time rendering system to reduce the overall primitive count. Figure 2 presents an overview of our method.

The core of our work is the perceptually inspired models, which aim to predict the visibility of distortions caused by the LOD selection. The factors responsible for the visibility of such changes include (1) the number of polygons and mesh complexity, (2) the apparent size of the shape, (3) the distance of the shape to the gaze position in visual angles (eccentricity), (4) position and type of the light source, (5) texture information, (6) object and background motion, and (7) average luminance of the display. Our models do not explicitly account for motion and display luminance. However, they are validated using dynamic scenes and can handle temporal artifacts due to rapid changes between LODs (Section 7). Furthermore, we study the visibility of distortions using only uniform backgrounds, thus minimizing the amount of perceptual data required for deriving the method. Although the presence of a background texture can reduce the visibility of missing geometrical details, our method remains conservative, as it does not try to account for visual masking from the background.

4. Spatial and temporal models for LOD

We model spatial and temporal effects in image space that can lead to visible distortions in LOD rendering. We use the spatial model to ensure that the simplified objects in the periphery appear the same as the original objects observed directly, i.e., in the fovea region. This enables the same perceived quality regardless of the eccentricity at which the object appears. Consequently, our spatial model, $M_S(\mathcal{R}_0, \mathcal{R}_i, ecc)$, takes as input the reference image $\mathcal{R}_0$ of an object without any mesh simplification and the test image $\mathcal{R}_i$ of the same object rendered using different LODs represented by the index i. $\mathcal{R}_0$ corresponds to the image of an original mesh shown in the foveal region, while $\mathcal{R}_i$ is the image of the simplified mesh shown at eccentricity ecc. The goal of the model is to predict the probability that an observer will detect the spatial differences introduced by LOD i at the specific eccentricity. Our temporal model is designed to handle popping artifacts when the LOD level changes when switching to a coarser or more detailed mesh. More precisely, given two images, $\mathcal{R}_i$ and $\mathcal{R}_{i+1}$, of the mesh rendered with subsequent LOD levels and the eccentricity ecc at which the switch between the two LOD occurs, the method predicts the probability, $M_T(\mathcal{R}_i, \mathcal{R}_{i+1}, ecc)$, that an observer detects popping artifacts. While the above metrics are taking 2D images as input, we show their application to 3D objects in Section 6.

4.1. Spatial model

The spatial model M_S separately accounts for geometrical distortions and shading distortions.

Geometrical distortions We assume that geometrical distortions are most visible around the boundaries of objects. We measure the magnitude of these distortions as the difference between the silhouettes of the objects presented in $\mathcal{R}_0$ and $\mathcal{R}_i$. To this end, we extract the silhouettes of the objects in both images as $\mathcal{S}(\mathcal{R}_0)$ and $\mathcal{S}(\mathcal{R}_i)$. For brevity, we denote them by $\mathcal{S}_0$ and $\mathcal{S}_i$, respectively, in the rest of the paper. In practice, silhouettes can be considered as sets of pixels that lie within the boundaries of the object in the image space. We measure the magnitude of the geometric distortion as the screen-space area of all pixels that belong to either silhouette but not both, i.e., area of $\mathcal{S}_0 \oplus \mathcal{S}_i$. To account for the lower sensitivity to distortions in the periphery, we utilize the theory of cortical magnification [CR74]. Instead of computing a simple area, we propose to compute a weighted area with weights decreasing as a function of eccentricity. Additionally, to account for area differences due to the size disparity between objects, we normalize the area by the perimeter of the reference silhouette. Consequently, we model the magnitude of the geometrical distortions as

$$G(\mathcal{R}_0, \mathcal{R}_i, ecc) = \frac{1}{\rho(\mathcal{S}_0)} \sum_{p \in \mathcal{S}_0 \oplus \mathcal{S}_i} w(ecc_p) \cdot A_p, \qquad (1)$$

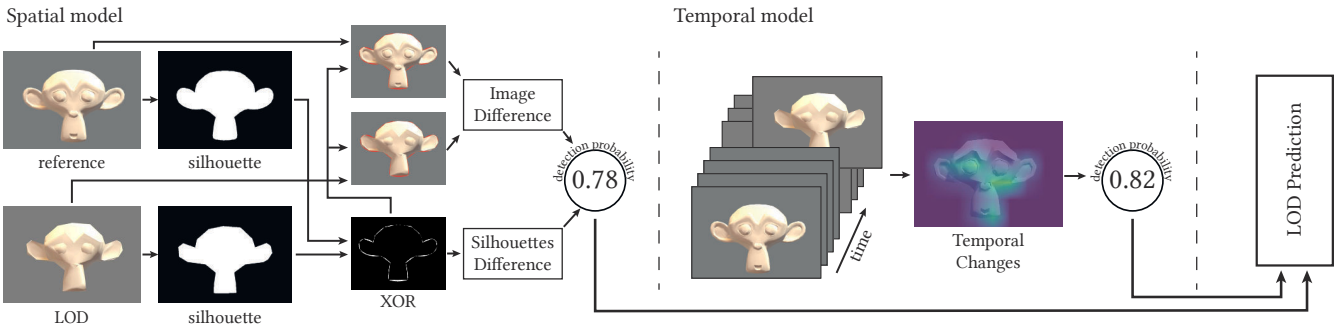


Figure 2: *Our technique for LOD selection uses two visibility models, spatial and temporal. The spatial (left) takes two images as input, one containing a rendering of the original mesh and the second containing the rendering of decimated (LOD) mesh. It separately computes the difference in silhouettes and the interior of the object. The combined difference is transformed into the probability of detecting spatial distortions. The task of the temporal model (middle) is to estimate the probability of detecting popping artifacts when the LOD changes, based on a short video sequence with the transition between two LODs.*

where p is an image pixel, ecc_p is the retinal eccentricity of the pixel p computed from ecc, A_p is the area of one pixel, $\rho(\mathcal{S}_0)$ is the perimeter of the silhouette $\mathcal{S}_0$ [Suz*85], and w is the weighting function. The selection of the weighting function w is critical for the performance of the model. Inspired by a softplus function, which has both exponential and linear parts and avoids negative values (Figure 3), we define our weighting function as

$$w(ecc_p) = \frac{1}{\beta} \cdot \log(1 + \exp(\alpha \cdot (ecc_p - s))) + b, \qquad (2)$$

where α, β, s, b are free parameters calibrated in Section 5.

Shading distortions Besides introducing geometrical distortions, low mesh quality can introduce inaccurate surface information during shading. To model the visibility of such distortions, we employ the state-of-the-art difference predictor FovVideoVDP [MDC*21]. Since we want to estimate only the distortions due to shading, before supplying the images $\mathcal{R}_0$ and $\mathcal{R}_i$ to FovVideoVDP, in both images we replace the $\mathcal{S}_0 \oplus \mathcal{S}_i$ region with a uniform gray background color. We later supply the two object images with the desired eccentricity value to the metric. The metric uses multiscale decomposition to model the visibility of changes, followed by a pooling strategy and conversion to quality values. To maintain compatibility with our measure for geometrical distortions, we seek an estimate of the distortion magnitude. Therefore, we omit the step that converts the pooling values from the multiscale decomposition into quality values. More precisely, we take the D_{pooled} value from the original technique [MDC*21, Section 3.8] (please see their paper for more details). In the rest of the paper, we refer to the predicted magnitude of the shading distortions from FovVideoVDP as $VDP(\mathcal{R}_0, \mathcal{R}_i, ecc)$.

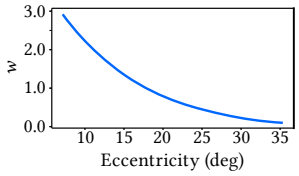


Figure 3: *The weighting function used for the eccentricity assigns lower weights to higher eccentricities (peripheral vision).*

Combined spatial model Our final model for spatial distortions combines models for geometric and spatial distortions. We treat both predictions as perceived magnitudes of distortions and apply Minkowski summation, which is often used in visual tasks to combine different cues [TLTT08]. Since the magnitudes reported by G and VDP models are valid up to scale, we additionally introduce scaling to the outcomes of the models before they undergo summation. Consequently, the estimated distortion magnitude after summation is given as

$$D(\mathcal{R}_0, \mathcal{R}_i, ecc) = \sqrt[k_s]{w_G(G(\mathcal{R}_0, \mathcal{R}_i, ecc))^{k_s} + w_{VDP}(VDP(\mathcal{R}_0, \mathcal{R}_i, ecc))^{k_s}}, \qquad (3)$$

where w_G, w_{VDP}, and k_s are the optimized parameters. Finally, to obtain the probability of detection, we use a sigmoid function as follows:

$$M_S(\mathcal{R}_0, \mathcal{R}_i, ecc) = \left(1 + e^{(-\zeta_s \cdot D(\mathcal{R}_0, \mathcal{R}_i, ecc) - \eta_t)}\right)^{-1}. \qquad (4)$$

The parameters w_G, w_{VDP}, k_s, ζ_s, η_s are fitted to the perceptual data by calibration (Section 5).

4.2. Temporal model

When using LOD techniques, a common problem is popping artifacts, i.e., temporal artifacts occurring when switching between two consecutive LOD levels, current (LOD_i) and the next one (LOD_{i+1}). To detect temporal artifacts, among the previously proposed models [TD22; KKW21], we adapt the one by Tursun et al. because it can readily process complex content. As input, our model takes two images, $\mathcal{R}_i$ and $\mathcal{R}_{i+1}$ corresponding to the renderings of consecutive LOD levels, i and $i+1$, at the eccentricity ecc where the objects are shown. Then it predicts the probability, $M_T(\mathcal{R}_i, \mathcal{R}_{i+1}, ecc)$, that an observer detects the transition. We assume the metric to be symmetric, i.e., the visibility of the change from LOD_i to LOD_{i+1} is the same in the opposite direction; more precisely, $M_T(\mathcal{R}_{i+1}, \mathcal{R}_i, ecc) = M_T(\mathcal{R}_i, \mathcal{R}_{i+1}, ecc)$. The temporal metric we adapt is designed to predict the local probability of detecting temporal changes in a video. However, we seek a pooled

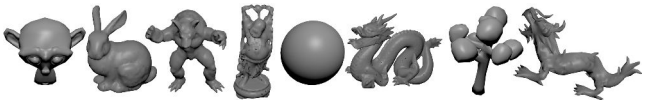


Figure 4: *Meshes used in our calibration experiments. From left: Suzanne, Bunny, Armadillo, Buddha, Sphere, Dragon, Tree, Asian Dragon.*

probability valid for the entire object. Similarly to how we adapted FovVideoVDP, we first extract from the metric the values before they are converted to probabilities, the JND-scaled spatio-temporal contrast C_M [TD22, Equation 10]. Then we apply our spatial pooling, after which the pooled values are converted to a probability using a sigmoid function. Before computing C_M, the images $\mathcal{R}_i$ and $\mathcal{R}_{i+1}$ must be converted to a temporal stimulus. In our work, we consider an instantaneous change between LOD levels. To futher mitigate the popping, another option is to fade in the two geometries and smooth transitioning between them. We do not apply this strategy because it requires expensive rendering of two LODs during the transition. Consequently, we transform the two images into a video of two frames, $V(\mathcal{R}_i, \mathcal{R}_{i+1}, ecc)$, with the images shown at eccentricity ecc. The final metric with custom pooling and sigmoid function is defined as

$$M_T(\mathcal{R}_i, \mathcal{R}_{i+1}, ecc) = \left(1 + e^{(-\zeta_t \cdot C - \eta_t)}\right)^{-1}, \tag{5}$$

where $C = \|C_M(V(\mathcal{R}_i, \mathcal{R}_{i+1}, ecc))\|_{k_t}$, the $\|\cdot\|_p$ is a p-norm. Additionally, ζ_t and η_t are the parameters of the sigmoid function. k_t, ζ_t, and η_t are free parameters fitted to the perceptual data in Section 5.

5. Calibration of the models

In this section, we describe the collection of perceptual data and the calibration of our models. In addition, we compare the performance of our spatial model with FovVideoVDP in an ablation study.

5.1. Calibration of the spatial model

For our spatial model, we optimize the set of free parameters $\theta = \{\alpha, \beta, s, b, w_G, w_{VDP}, k_s, \zeta_s, \eta_s\}$ using perceptual experiment data containing images with different amounts of distortions of the LOD and the corresponding probabilities of detecting these distortions at different eccentricities.

Stimuli We used a set of 8 meshes (Figure 4) from Stanford 3D Scanning Repository [Sta23], Blender 3D [Ble], and Unity Asset Store [Uni23]. For each mesh, we computed four geometries corresponding to four LOD levels using the method of Garland and Heckbert [GH97]. Each of the resulting geometries was rendered in three different sizes, spanning between 1.22 and 12.22 visual degrees on the screen. We considered two different environment maps for lighting: one captures the indoor illumination of a chapel [Dim22a], while the other one captures the outdoor lighting conditions from a landscape [Dim22b]. All objects were rendered in uniform white color and semi-matte appearance.

Task The experiment was carried out using the two alternative forced choice (2AFC) procedure. In each trial, the participants were shown three images. The reference image of the original mesh was shown in the center, while the two test images were shown on either side at the same eccentricity. One of the test images was obtained by rendering an exact copy of the reference, while the other was an LOD geometry. The left/right position of the test image was randomized in each trial, and one of the images was flipped around the horizontal axis to keep the objects symmetrical with respect to the center of the screen. A small cross was shown at the center as the fixation target. The gaze position was monitored using an eye tracker, and to avoid viewing the objects at a different eccentricity, e.g., due to involuntary eye movements, the stimuli were hidden if the gaze position deviated from the fixation target. Participants were asked to use the arrow keys on the keyboard to select the image that looked more similar to the reference in the center.

The test images were shown at three eccentricities: 7.88°, 15.47°, and 22.56°. For the smallest eccentricity, the largest size of the objects was not used due to the overlap with the reference image. In total, 256 different stimuli were shown (8 meshes $\times$ 4 LODs $\times$ 3 sizes $\times$ 3 eccentricities - 1 size $\times$ 8 meshes $\times$ 4 LODs) and each trial was repeated twice. In total, each participant performed 512 trials. The order of the stimuli was randomized to avoid bias.

Hardware We used a Tobii Pro Spectrum eye tracker at 600 Hz connected to a 27" Acer Predator display at 3840×2160 resolution and a refresh rate of 120 Hz with a peak luminance of 170 cd/m^2.

Participants 15 people (3 from authors and 12 naive to the field) with normal or corrected-to-normal vision participated in the experiment (ages 20–39, mean: 26.6).

Optimization The training dataset from our experiment is a set of tuples

$$\mathcal{D} = \{\langle \mathcal{R}_0, \mathcal{R}_i, ecc_j, v_k \rangle\}, \tag{6}$$

where $\mathcal{R}_0$ is the image of the reference object, $\mathcal{R}_i$ is the image of a simplified LOD, and ecc_j is the stimulus eccentricity. v_k is a binary variable that represents a response in the experiment and takes the value of 1 if the observer detects visual distortions from mesh simplification (i.e., selects the reference image instead of LOD) and 0, otherwise. Using our dataset $\mathcal{D}$, we compute the probability, $P(\mathcal{R}_0, \mathcal{R}_i, ecc_j)$, that an observer detects the mesh simplification as the mean of corresponding v_k in the dataset, which aggregates binary responses across participants. Then, we estimate the set of optimal parameter values, θ^* by minimizing the function

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \sum_{\mathcal{D}} \|M_S(\mathcal{R}_0, \mathcal{R}_i, ecc_j)_\theta - P(\mathcal{R}_0, \mathcal{R}_i, ecc_j)\|^2. \tag{7}$$

We report the optimal parameters in Table 1.

Validation We validate our model by evaluating the correlation and mean absolute error (MAE) between the predicted probabilities given by Equation 4 and $P(\mathcal{R}_0, \mathcal{R}_i, ecc_j)$. In a 4-fold cross-validation, the Pearson correlation coefficient is 0.74 ± 0.051 and the mean absolute error (MAE) is 0.10 ± 0.006 (mean $\pm$ standard error of the mean across folds). The Pearson correlation coefficient is 0.81 and the MAE is 0.09 when the model is trained on the entire dataset.

5.2. Calibration of the temporal model

We optimize the set of free parameters $\{k_t, \zeta_t, \eta_t\}$ of our temporal model in a similar way to the spatial model.

Stimuli We tested 5 of the same 8 meshes (Figure 4) at eccentricities 7.88°, 15.47°, and 22.56°, with four LODs generated by the progressive mesh technique [Hop96]. We considered 8 repetitions per stimulus, and each trial showed one LOD transition.

Task Initially, two instances of the same object at the same LOD were randomly placed on either side of the screen at the same eccentricity. One of the instances was displayed without any temporal change, while the other switched to the next simpler LOD after 0.5 s. After one second, the objects were hidden to prevent the participant from perceiving spatial differences. In each trial, we asked the participant to select the side (left/right) where they perceived a temporal change by pressing a key on the keyboard. We used the same hardware as in Section 5.1.

Participants Five people (three of whom are authors of the paper) participated in the study. All of them had normal or corrected to normal vision.

Optimization The parameter set is calibrated in the same way as the spatial model (see Equation 7). The optimal parameters are given in Table 1.

Validation According to the correlation of the predicted probabilities derived from optimized pooling with the measured probabilities of the perceptual experiment, the Pearson correlation coefficient is 0.91, and the mean absolute error is 0.06 on the entire dataset. We compared our correlation with the Tursun model et al. in its original formulation [TD22], optimizing for the parameter β, responsible for the pooling of the probability map. It resulted in a poorer performance than our model (Pearson correlation coefficient $= 0.53$, $MAE = 0.18$ with a $\beta = 3.24$). We also performed a leave-two-out cross-validation for our method, for which the Pearson correlation coefficient is 0.86 ± 0.016, and the mean absolute error is 0.08 ± 0.006 (mean $\pm$ SEM across folds).

Table 1: *The optimized parameters of the spatial and temporal models.*

α	β	s	b	w_G	w_{VDP}
-9.41	$7.02 \cdot 10^{-3}$	$-2.26 \cdot 10^{-2}$	$3.43 \cdot 10^{-2}$	2.84	0.42
k_s	ζ_s	η_s	k_t	ζ_t	η_t
5.39	$6.99 \cdot 10^{-3}$	$8.12 \cdot 10^{-3}$	1.74	0.08	0.21

5.3. Ablation study

The proposed combined spatial model of geometrical and shading distortions provides the best performance in validation. To measure the contribution of having a separate component trained on silhouette differences, we compare the predictions of our method with the predictions of FovVideoVDP alone. We compute FovVideoVDP predictions on images of the reference object $\mathcal{R}_0$ and its LOD, $\mathcal{R}_i$. However, different from the shading component of our model, in the ablation study, we do not remove object boundaries computed from silhouette differences while rendering the input images.

Given the output intensities D_{pooled} of $VDP(\mathcal{R}_0, \mathcal{R}_i, ecc)$ for eccentricity ecc, we compute quality scores in just objectionable difference (JOD) units using Equation 19 from Mantiuk et al. [MDC*21]:

$$Q_{\text{JOD}} = 10 - \alpha_{\text{JOD}}(D_{\text{pooled}})^{\beta_{\text{JOD}}}. \tag{8}$$

We compute the predicted probabilities from the quality scores Q_{JOD} using the conversion of Perez-Ortiz and Mantiuk [PM17, Equation 5]. FovVideoVDP was originally calibrated on its training dataset. To avoid dataset bias and make a fair comparison, we recalibrate FovVideoVDP predictions by fine-tuning parameters α_{JOD} and β_{JOD} to minimize the mean absolute error with our experiment data.

The results of this ablation study are shown in Figure 5, where we provide the predictions using our method (Equation 4) and FovVideoVDP. We plot both predictions against the ground truth, $\{P(R_i, T_i, ecc_i)\}$ (Section 5). Overall, we observe a higher Pearson correlation (PCC) between our combined model and ground truth (0.81) compared to FovVideoVDP and ground truth (0.72). The result of cross-validation on individual folds is reported in Table 2. This study shows that the use of FovVideoVDP alone results in poorer performance, and having a separate model trained for the effect of silhouette distortions on visibility results in more accurate predictions.

Score	f#1	f#2	f#3	f#4	avg	all
MAE FovVideoVDP	0.10	0.10	0.09	0.09	**0.10**	**0.09**
MAE Ours	0.10	0.11	0.08	0.10	**0.10**	**0.08**
PCC FovVideoVDP	0.68	0.62	0.70	0.74	**0.69**	**0.72**
PCC Ours	0.76	0.60	0.74	0.85	**0.74**	**0.81**

Table 2: *4-fold Cross-validation for our method and FovVideoVDP. The data from a certain mesh is used either in training or in the validation set.*

6. Gaze-contingent LOD prediction

We use our spatial and temporal models in a new method to select an optimal LOD in gaze-contingent rendering. For a given object, the input to the method is a set of meshes corresponding to different LODs. We denote these by LOD_i for $0 \leq i \leq N$, where LOD_0 and LOD_N correspond to the highest and lowest quality meshes, respectively. The goal of the method is to find the optimal mapping, L, from the pairs (d, ecc) to the highest LOD index i that does not lead to visible artifacts, where d is the distance to the rendered object and ecc is the eccentricity at which it is rendered.

To define the mapping $L(d, ecc)$, let us first consider what a good LOD candidate is to be shown at a distance d and eccentricity ecc. First, using $LOD_{L(d,ecc)}$ cannot lead to any visible spatial distortions compared to the rendering of the original mesh, LOD_0. By denoting a rendered image with LOD_i by $\mathcal{R}_i$, we can write this

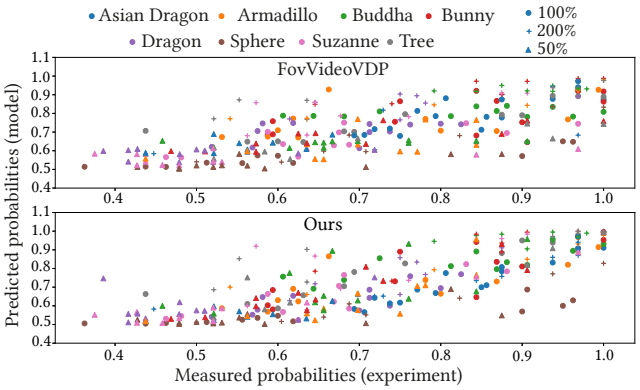


Figure 5: *The plots of probability predictions with our method (bottom) and FovVideoVDP [MDC*21] (top). We observe a higher level of correlation between our predictions and the ground truth compared to FovVideoVDP. The percentages indicate the different sizes of the mesh.*

condition using our spatial model as

$$M_S(\mathcal{R}_0, \mathcal{R}_{L(d,ecc)}, ecc) < \delta, \tag{9}$$

where δ is the predefined detection probability threshold below which we consider the distortions to be insignificant. In our experiments, we set $\delta = 0.75$. This choice is motivated by the common definition of threshold, which is the midpoint between seeing and not seeing distortions. The second condition is to avoid visible temporal changes when the LODs change. It is important to note that this condition is only critical for (d, ecc) corresponding to the LOD temporal changes. Moreover, due to the symmetry of the metric (Section 4.2), we can limit our consideration to the temporal changes due to switching from a lower level. Hence, the second condition using our temporal model is given as

$$M_T(\mathcal{R}_{L(d,ecc)-1}, \mathcal{R}_{L(d,ecc)}, ecc) < \delta. \tag{10}$$

Since the evaluation of both conditions in real-time scenarios would be prohibitively expensive, we propose to precompute the mapping L and store it in a look-up table (LUT). To extend the mapping to different object orientations, we evaluate the conditions for a set of K orientations. We denote the rendering of the rotated LOD as $\mathcal{R}_i^k$, for $1 \leq k \leq K$. Finally, we can define the optimal mapping L as an optimization that aims to maximize i (i.e., the coarsest LOD_i) satisfying the conditions above for a given eccentricity and distance, and taking the lowest LOD across different rotations to make the mapping conservative. Formally, we can define as

$$L(d, ecc) = \min_{1 \leq k \leq K} \max_{0 \leq i \leq N} \quad i \tag{11}$$

$$\text{s.t.} \quad M_S\left(\mathcal{R}_0^k, \mathcal{R}_i^k, ecc\right) < \delta, \quad M_T\left(\mathcal{R}_{i-1}^k, \mathcal{R}_i^k, ecc\right) < \delta. \tag{12}$$

In practice, we precompute the mapping L by densely sampling the eccentricities, distances, as well as orientations, and LOD levels of the object. The values are later stored in the LUT for rendering, where the optimal LOD can be obtained instantaneously with a conservative nearest-neighbor search between the entries. The time of precomputation is approximately 6 seconds per one metric evaluation with unoptimized code.

6.1. Contributions of the spatial and temporal models

To gain insights on how the spatial and temporal models contribute to the final predictions, we analyze the predicted detection probabilities when changing the number of LODs between a fixed highest quality and lowest quality meshes. We remind that the objective of the spatial component is to predict a probability of seeing spatial distortions, while the objective of the temporal component is to mitigate or eliminate the popping artifacts. Therefore, in theory, the higher the number of LODs between a reference mesh and the corresponding lowest quality mesh, the less the temporal changes will be visible. To put into practice, we analyzed the same object simplified to a different number of LODs (Figure 6). The temporal model of the Sphere with 13 LODs has less impact than the spatial, while the contribution is reversed when using only four LODs. To confirm the generality of this observation, we recomputed all our predictions increasing the number of LODs for each mesh from four to 13. When using four LODs, the choice of the optimal LOD was driven by the spatial model in 16% of the stimuli and by the temporal model in 84%, indicating that the temporal changes are more visible than the spatial differences. When using 13 LODs, in 41% of stimuli the spatial component is dominant for the choice of the optimal LOD.

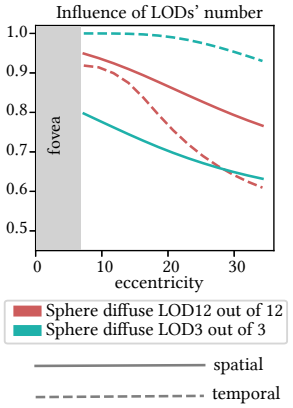


Figure 6: *Changes in the probability of detection estimates from the spatial and temporal models of our method. In the plot, we show the effect of having different numbers of discrete mesh simplification levels used in rendering.*

7. Evaluation

We evaluate our method in a gaze-contingent scenario and provide a comparison with a recent industrial solution for mesh simplification in Unreal Engine called Nanite [Epi23].

7.1. Application to gaze-contingent rendering

Our method for LOD prediction (Section 6) can be adapted into gaze-contingent rendering systems allowing not only reducing mesh quality for distant objects but also for those which appear in the periphery.

Rendering application We tested our prediction technique for LOD on two scenes (Figure 7). The first one was CATHEDRAL with 16 test objects placed on the sides of the main aisle (see suppl. video). The objects were always randomly chosen from a set of 5 geometries, different from those used for our method's calibration. Each object was rendered with a different material (i.e., gloss

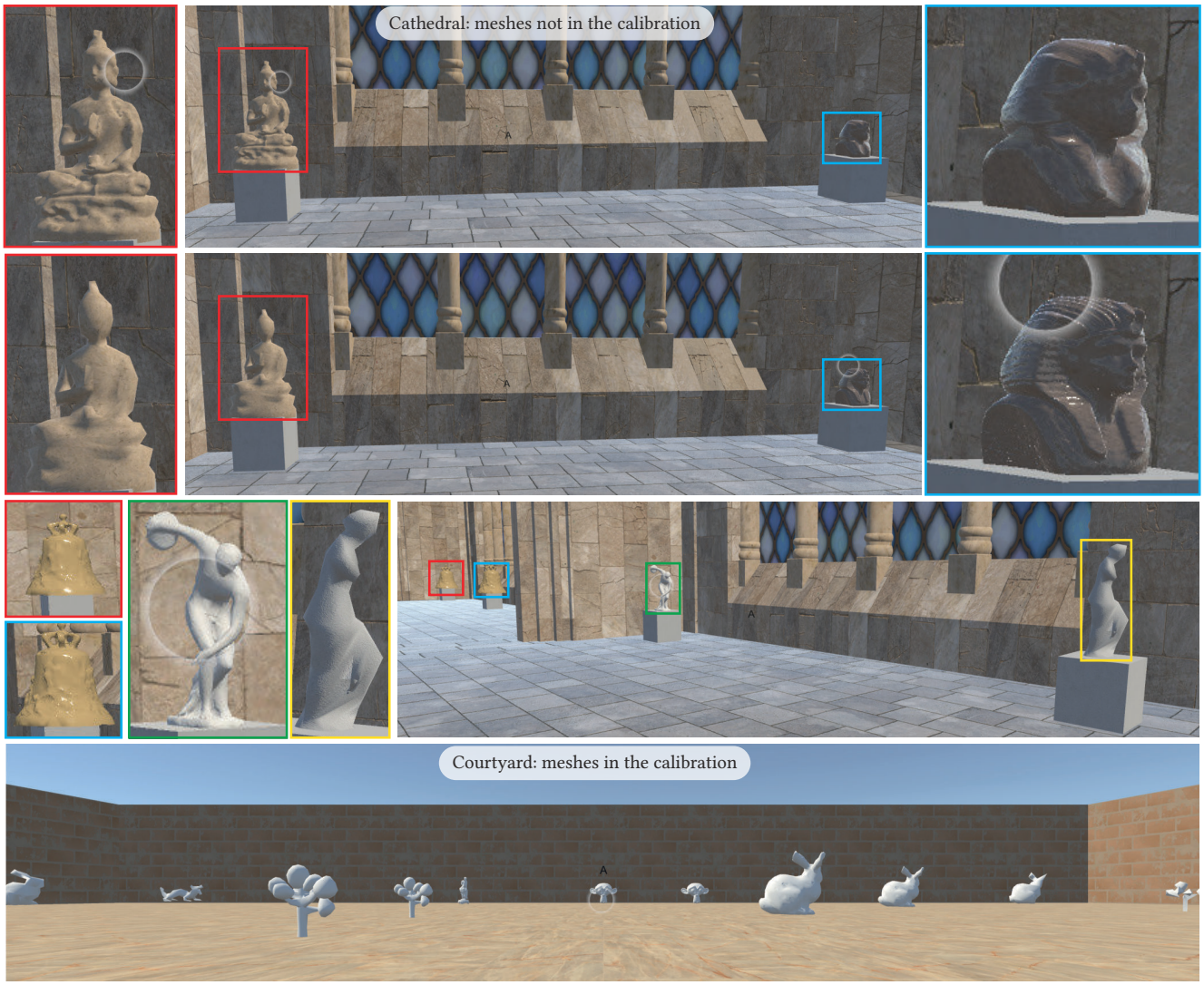


Figure 7: *The first three rows of the figure show two frames taken from the application (scene* CATHEDRAL*). In the first frame, the gaze is located on the idol statue (left). In the second frame, the gaze is located on the pharaoh (right). According to the eccentricity of the object, the rendering shows different LODs. In the third row, the gaze is located on the discobolus (third object from left). The bell is rendered with two different LODs, the discobolus with the maximum LOD. The fourth row shows a frame of the scene* COURTYARD.

and texture), and the scene was rendered with illumination represented by an environment map resembling the light conditions in the cathedral [Dim22a]. The second scene was a COURTYARD illuminated with a directional light source. The simple geometry of the scene was textured with wood and brick textures. The skybox was used as a background. On the floor, we placed 40 randomly drawn objects from the eight meshes previously used in the calibration. While some objects were static, others were moving on a circular trajectory. The objects in this scene were rendered with a semi-matte appearance without any texture.

We used the same strategy for both scenes to precompute the LUT for the LOD prediction. We uniformly sampled 4 distances and 13 eccentricities that span the entire range necessary for the experiment. Since both scenes were always observed from the same height, we considered rotations of the objects only around the vertical axis with 30° steps. We did not consider orientations as a dimension of LUT. Instead, we store one value which accounts for the worst case among all orientations. Therefore, for each mesh our LUT stores optimal LOD for 4 different distances and 13 eccentricities, i.e., 4 x 13 = 52 integer values. Both scenes were tested on a desktop screen (same as for our calibration experiment) and an HTC Vive Pro Eye VR headset, both equipped with an eye tracker.

Study 10 participants (mean age: 26.9, all naive to the study, with normal or corrected-to-normal vision) took part in our experiment. After a brief introduction and eye tracker calibration, participants were asked to start the experiment. We also explained the concept

Table 3: *The table reports the preference for our method for both scenes and display setups with p-values from a binomial test shown in parentheses. The last two columns show the average triangle counts of the two visualization modes.*

Scene	VR	Desktop	#△ Ours	#△ Orig.
Cathedral	40% (0.38)	60% (0.83)	53698	98832
Courtyard	50% (0.62)	40% (0.38)	13339	44363

of popping artifacts to the participants so they took it into account in their evaluation. Each participant was exposed to both scenes using VR and desktop setups. Participants could freely move for the CATHEDRAL scene, while in COURTYARD, the viewing position was fixed, and participants could only rotate the camera/head. The participants remained seated throughout the experiment, and camera movement was possible using keyboard keys. For each scene and viewing condition, participants could switch with a single key between two visualization modes: the first using our LOD prediction and the second using original objects without mesh simplification. After a one-minute exploration stage, participants were asked which version had higher visual quality. The order of the stimuli and the scenes was randomized.

Results Table 3 shows the results of our evaluation. The *p*-values (binomial test) are given in parentheses. The results do not indicate a substantial perceptual difference between the two methods, suggesting that our method provides a quality similar to the original rendering. The table also compares the polygon counts between the two visualization modes for a one-minute first-person navigation along a predefined path, where the gaze position was simulated in the center of the screen. Our method can lead to further computational savings as the geometric complexity of objects increases.

7.2. Comparison with Nanite

We compare the performance of our technique with that of Nanite, a state-of-the-art LOD system developed for Unreal Engine [Epi23]. The goal of the test is not to demonstrate superiority over Nanite, but to express where our method stands in relation to a complex solution like Nanite and also how the two techniques perform together. Our motivation is that geometry can still be an issue in the absence of highly optimized techniques such as Nanite.

Unlike our method, Nanite does not take eccentricity into account. It reduces the number of rendered polygons based on the relation between their screen-space size and the pixel size. To our knowledge, it does not account for any perceptual effects and scales roughly linearly with the screen's resolution.

The test was conducted on a system with NVidia GeForce RTX 3090, Intel i9-12900K 3.20 GHz and 128 GB of memory. We used Unreal Engine 5.1 to render a scene containing copies of a reference object placed on a uniform 3D grid. We tested the rendering performance at 1k = 1024x540, 2k = 2048x1080, 4k = 4096x2160 resolutions with two reference objects in four different configurations, namely, NANITE ONLY, OURS ONLY, NANITE+OURS and ALL OFF. The NANITE ONLY configuration rendered the scene with Nanite using only the reference mesh, while the NANITE+OURS configuration rendered the scene with Nanite in combination with our method using the same set of LODs as OURS ONLY. The ALL OFF configuration rendered the scene with neither optimization scheme, using only the reference mesh. The test results in average frames per second (FPS) are given in Figure 8 and sample images from the test are shown in Figure 9. Also, a separate supplemental video provides captures of the sample test runs along with a sequence illustrating the visualization of our method.

2184 copies of the reference object were used in each scene, with a total raw triangle count of 2.20 and 2.36 billion for the Suzanne and Asian Dragon reference meshes, respectively. The two reference meshes are of higher quality than those used in the previous experiments (Table 3) since Nanite yields negligible gain with low resolution meshes. In particular, we used a finely detailed version of the Asian Dragon object and subdivided the original Suzanne mesh using the Catmull-Clark algorithm, resulting in a much smoother mesh than the original.

The ALL OFF configuration was rendered at less than 10 FPS for all runs. It can be seen that the OURS ONLY configuration provided a rendering performance that is comparable to Nanite. The NANITE+OURS configuration, where our method was used together with Nanite, consistently improved the rendering performance over the NANITE ONLY configuration. Such that, NANITE+OURS achieved a maximum average speedup of 33% at 2K resolution and 24% at 4K resolution compared to NANITE ONLY. These results show that our relatively simple technique based on foveation can provide additional benefits when applied on top of a complex technique such as Nanite.

8. Limitations and future work

To keep the duration of our subjective experiment sessions feasible, we decided to limit the range of appearance properties considered in the calibration. In particular, we used simple shading on semi-matte objects without non-local shadows and reflections. Similarly, we did not calibrate our method using a diverse set of textures or consider the influence of background textures on spatial, temporal, and silhouette distortions. However, our method shows promise in extending to different surface properties, including a selected set of simple textures shown in Figure 10, since we rely on a general-purpose image quality metric, FovVideoVDP. We expect that the LODs selected by our method will not result in visible artifacts in the presence of complex textures because textures on an object are likely to act as visual maskers and hide potential artifacts. Nevertheless, a validation on an extensive texture dataset, including high-contrast textures and other effects related to textures remains as a future work.

Our model does not take into account different levels of luminance contrast between the object and the background. We calibrate our method for the worst-case scenario in the luminance contrast setting with bright objects on a dark background, which results in high visibility of changes to the object silhouette. We also did not consider depth perception in stereo viewing, where oversimplified geometries increase the visibility of mesh simplification due to a flat appearance that lacks depth variations of the original geometry.

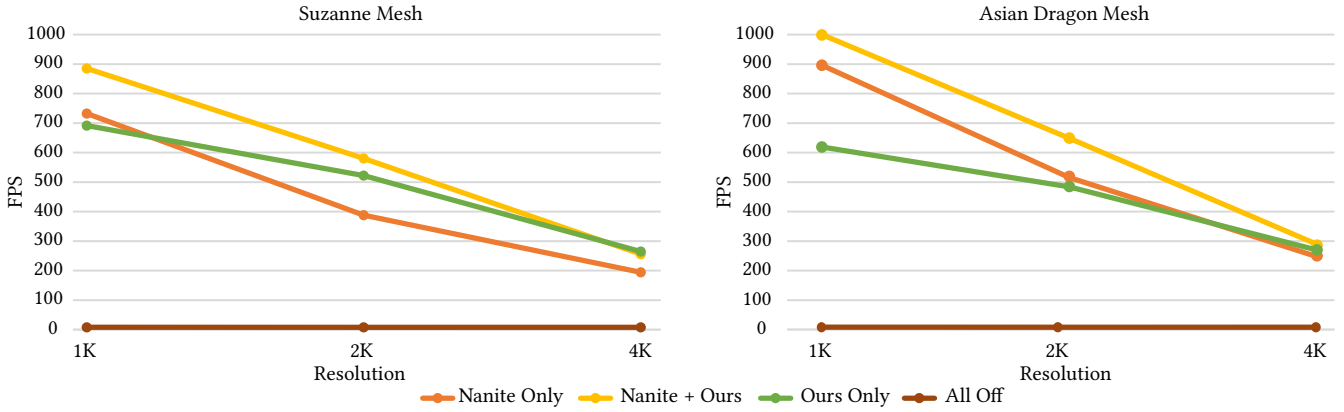
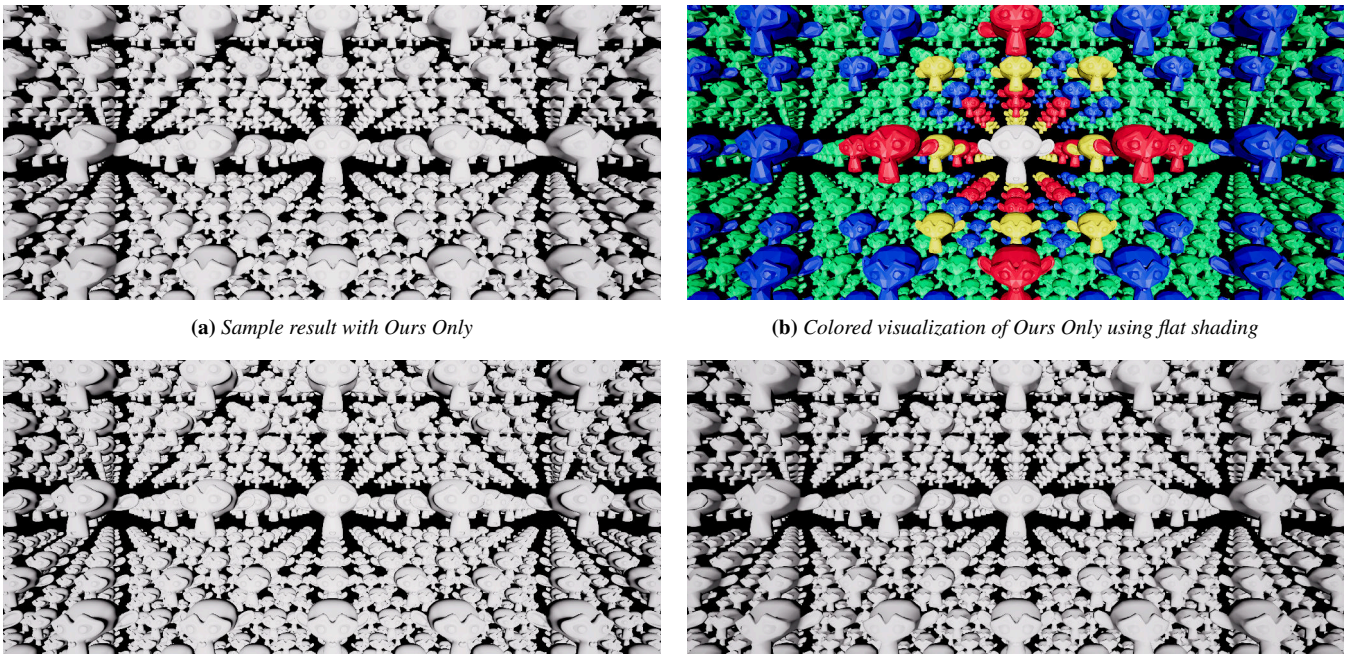


Figure 8: *Results of the performance comparison test for the four different configurations.*



(a) *Sample result with Ours Only*

(b) *Colored visualization of Ours Only using flat shading*

(c) *Sample result with Nanite Only*

(d) *Sample result with Nanite+Ours*

Figure 9: *Sample images from the performance comparison test using the Suzanne mesh. The gaze position was simulated at the screen center. In (b), the LODs selected by our method are visualized in white, yellow, red, blue and green in descending order of detail, s.t., those in white were rendered using the reference mesh, while those in green were rendered using the simplest LOD. The* NANITE ONLY *run used only the reference mesh, while the* NANITE+OURS *run used the same set of LODs as* OURS ONLY.

Furthermore, our method also does not explicitly handle the effects of illumination, although our LOD selection strategy can account for different illumination conditions when precomputing the LUT.

Our calibration is based on LODs generated by a single technique [Hop96]. Although our prediction model does not directly exploit any property of this technique that would hinder generalization, extension to other LOD methods may be investigated further in the future.

Currently, our method is applicable only to geometries representing individual objects with clear boundaries within the field of view. To handle large meshes, such as foliage or terrain, several extensions are required. First, the computation of silhouettes should be updated to also take into account internal silhouettes that arise in the presence of self-occlusion, as often observed in terrain geometries. Second, the LOD prediction method should predict a spatially-varying map of allowed mesh degradation instead of a single LOD selection for the whole geometry. After including those extensions, our method requires further benchmarking on complex scenes.

Figure 10: *The textures used on the objects of cathedral scene during the validation experiment. From left: metal, plaster, marble, marble stone, stucco.*

Although addressing all these limitations provides exciting directions for future work, our present experiments demonstrate good performance and the benefits of our method.

9. Conclusion

Foveated rendering is considered a key enabler for novel virtual and augmented reality headsets. With the rendering quality better aligned with the requirements of the human visual system, the rendering system can become more efficient or be able to render more complex and immersive environments. While a lot of recent research effort was dedicated to foveated rendering methods that reduce spatial resolution or shading rate, the cost of processing complex geometries is equally important. This work proposes a simple yet efficient way to predict LOD suitable for rendering an object at a specific eccentricity. The method can be easily combined with existing real-time rendering engines and immediately exploited to provide performance boosts. We believe that by building upon our model and observations, future work can extend our model to more accurately model the effects of texture and illumination, as well as handle different types of geometries.

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement N○ 804226 PERDY).

References

[AČMS10] AYDIN, TUNÇ OZAN, ČADÍK, MARTIN, MYSZKOWSKI, KAROL, and SEIDEL, HANS-PETER. "Visually Significant Edges". *ACM Trans. Appl. Percept.* 7.4 (July 2010). ISSN: 1544-3558. DOI: 10.1145/1823738.1823745 3.

[BC88] BROWDER, G. and CHAMBERS, W. "Eye-slaved area-of-interest display systems". *Flight Simualtion Technologies Conference*. 1988, 4636 2.

[Ble] BLENDER. URL: https://www.blender.org 5.

[CD96] CORTESE, JAMES M and DYRE, BRIAN P. "Perceptual similarity of shapes generated from Fourier descriptors." *Journal of Experimental Psychology: Human Perception and Performance* 22.1 (1996), 133 2.

[CDS*22] CHEN, SHAOYU, DUINKHARJAV, BUDMONDE, SUN, XIN, et al. "Instant Reality: Gaze-Contingent Perceptual Optimization for 3D Virtual Reality Streaming". *IEEE Transactions on Visualization and Computer Graphics* 28.5 (2022), 2157–2167 2.

[CLZ01] CHANG, CHENG, LIU, WENYIN, and ZHANG, HONGJIANG. "Image retrieval based on region shape similarity". *Storage and Retrieval for Media Databases 2001*. Vol. 4315. International Society for Optics and Photonics. 2001, 31–38 2.

[CR74] COWEY, A and ROLLS, ET. "Human cortical magnification factor and its relation to visual acuity". *Experimental Brain Research* 21.5 (1974), 447–454 3.

[DÇ07] DUCHOWSKI, ANDREW T and ÇÖLTEKIN, ARZU. "Foveated gaze-contingent displays for peripheral LOD management, 3D visualization, and stereo imaging". *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 3.4 (2007), 1–18 2.

[Dim22a] DIMITRIOS SAVVA, JAROD GUEST. *Thatch Chapel*. 2022. URL: https://polyhaven.com/a/thatch_chapel 5, 8.

[Dim22b] DIMITRIOS SAVVA Greg Zaal, JAROD GUEST. *Belfast Sunset*. 2022. URL: https://polyhaven.com/a/belfast_sunset_puresky 5.

[Duc02] DUCHOWSKI, ANDREW T. "A breadth-first survey of eye-tracking applications". *Behavior Research Methods, Instruments, & Computers* 34.4 (2002), 455–470 2.

[Epi23] EPICGAMES. *Unreal Engine 5.1 Documentation - Nanite*. Accessed: Jan-2023. 2023. URL: https://docs.unrealengine.com/5.1/en-US/nanite-virtualized-geometry-in-unreal-engine 2, 7, 9.

[FS93] FUNKHOUSER, THOMAS A. and SÉQUIN, CARLO H. "Adaptive display algorithm for interactive frame rates during visualization of complex virtual environments". *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. 1993, 247–254 2.

[GFD*12] GUENTER, BRIAN, FINCH, MARK, DRUCKER, STEVEN, et al. "Foveated 3D graphics". *ACM Transactions on Graphics (TOG)* 31.6 (2012), 1–10 2.

[GH97] GARLAND, MICHAEL and HECKBERT, PAUL S. "Surface simplification using quadric error metrics". *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. 1997, 209–216 5.

[Gle94] GLENN, WILLIAM E. "Real-Time Display Systems". *Visual Science and Engineering: Models and Applications* (1994), 387 2.

[HM93] HITCHNER, LEWIS E. and MCGREEVY, MICHAEL W. "Methods for user-based reduction of model complexity for virtual planetary exploration". *Human Vision, Visual Processing, and Digital Display IV*. Vol. 1913. SPIE. 1993, 622–636 3.

[HML*21] HASSELGREN, JON, MUNKBERG, JACOB, LEHTINEN, JAAKKO, et al. "Appearance-Driven Automatic 3D Model Simplification". *Eurographics Symposium on Rendering*. 2021 2.

[Hop96] HOPPE, HUGUES. "Progressive meshes". *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. 1996, 99–108 6, 10.

[Jac95] JACOB, ROBERT JK. "Eye tracking in advanced interface design". *Virtual environments and advanced interface design* 258 (1995), 288 2.

[KG82] KUHL, FRANK P and GIARDINA, CHARLES R. "Elliptic Fourier features of a closed contour". *Computer graphics and image processing* 18.3 (1982), 236–258 2.

[KG96] KORTUM, PHILIP and GEISLER, WILSON S. "Implementation of a foveated image coding system for image bandwidth reduction". *Human Vision and Electronic Imaging*. Vol. 2657. International Society for Optics and Photonics. 1996, 350–360 2.

[KKW21] KRAJANCICH, BROOKE, KELLNHOFER, PETR, and WETZSTEIN, GORDON. "A perceptual model for eccentricity-dependent spatio-temporal flicker fusion and its applications to foveated graphics". *ACM Transactions on Graphics (TOG)* 40.4 (2021), 1–11 4.

[KSL*19] KAPLANYAN, ANTON S, SOCHENOV, ANTON, LEIMKÜHLER, THOMAS, et al. "DeepFovea: Neural reconstruction for foveated rendering and video compression using learned statistics of natural videos". *ACM Transactions on Graphics (TOG)* 38.6 (2019), 1–13 2.

[LL00] LATECKI, LONGIN JAN and LAKAMPER, ROLF. "Shape similarity measure based on correspondence of visual parts". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.10 (2000), 1185–1190 2.

[Lof08] LOFFLER, GUNTER. "Perception of contours and shapes: Low and intermediate stage mechanisms". *Vision research* 48.20 (2008), 2106–2127 2.

[LRC*03] LUEBKE, DAVID, REDDY, MARTIN, COHEN, JONATHAN D, et al. *Level of detail for 3D graphics*. Morgan Kaufmann, 2003 2.

[MD01] MURPHY, HUNTER and DUCHOWSKI, ANDREW T. "Gaze-Contingent Level Of Detail Rendering". *Eurographics 2001 - Short Presentations*. Eurographics Association, 2001. DOI: 10.2312/egs.20011004 2, 3.

[MDC*21] MANTIUK, RAFAŁ K, DENES, GYORGY, CHAPIRO, ALEXANDRE, et al. "FovVideoVDP: A visible difference predictor for wide field-of-view video". *ACM Transactions on Graphics (TOG)* 40.4 (2021), 1–19 2, 4, 6, 7.

[MIGS22] MOHANTO, BIPUL, ISLAM, ABM TARIQUL, GOBBETTI, ENRICO, and STAADT, OLIVER. "An integrative view of foveated rendering". *Computers & Graphics* 102 (2022), 474–501 2.

[MKRH11] MANTIUK, RAFAŁ, KIM, KIL JOONG, REMPEL, ALLAN G., and HEIDRICH, WOLFGANG. "HDR-VDP-2: A Calibrated Visual Metric for Visibility and Quality Predictions in All Luminance Conditions". *ACM Trans. Graph.* 30.4 (July 2011). ISSN: 0730-0301. DOI: 10.1145/2010324.1964935 2.

[MPR08] MATA, SUSANA, PASTOR, LUIS, and RODRÍGUEZ, ÁNGEL. "Mesh Simplification using Distance Labels for View-Independent Silhouette Preservation". *3rd International Conference on Computer Graphics Theory and Applications (GRAPP 2008)*. Ed. by J. BRAZ, N. JARDIM and MADEIRAS, J. Eurographics. Portugal: INSTICC, Jan. 2008, 5–14 3.

[OYT96] OHSHIMA, TOSHIKAZU, YAMAMOTO, HIROYUKI, and TAMURA, HIDEYUKI. "Gaze-directed adaptive rendering for interacting with virtual space". *Proceedings of the IEEE 1996 Virtual Reality Annual International Symposium*. IEEE. 1996, 103–110 2, 3.

[PKS*16] PATNEY, ANJUL, KIM, JOOHWAN, SALVI, MARCO, et al. "Perceptually-based foveated virtual reality". *ACM SIGGRAPH 2016 emerging technologies*. 2016, 1–2 2.

[PLL06] PARK, BO GUN, LEE, KYOUNG MU, and LEE, SANG UK. "A new similarity measure for random signatures: Perceptually modified Hausdorff distance". *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer. 2006, 990–1001 2.

[PM17] PEREZ-ORTIZ, MARIA and MANTIUK, RAFAL K. "A practical guide and software for analysing pairwise comparison experiments". *arXiv preprint arXiv:1712.03686* (2017) 6.

[PSK*16] PATNEY, ANJUL, SALVI, MARCO, KIM, JOOHWAN, et al. "Towards foveated rendering for gaze-tracked virtual reality". *ACM Transactions on Graphics (TOG)* 35.6 (2016), 1–12 2.

[Red96] REDDY, MARTIN. "SCROOGE: Perceptually-Driven Polygon Reduction". *Computer Graphics Forum*. Vol. 15. 4. Wiley Online Library. 1996, 191–203 3.

[RLMS03] REINGOLD, EYAL M, LOSCHKY, LESTER C, MCCONKIE, GEORGE W, and STAMPE, DAVID M. "Gaze-contingent multiresolutional displays: An integrative review". *Human factors* 45.2 (2003), 307–328 2.

[SHI96] SHUM, HEUNG-YEUNG, HEBERT, MARTIAL, and IKEUCHI, KATSUSHI. "On 3D shape similarity". *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE. 1996, 526–531 2.

[SIK*16] SWAFFORD, NICHOLAS T., IGLESIAS-GUITIAN, JOSÉ A., KONIARIS, CHARALAMPOS, et al. "User, Metric, and Computational Evaluation of Foveated Rendering Methods". *Proceedings of the ACM Symposium on Applied Perception*. SAP '16. Anaheim, California: Association for Computing Machinery, 2016, 7–14. ISBN: 9781450343831. DOI: 10.1145/2931002.2931011 2.

[Sta23] STANFORD. *3D Scanning Repository*. Accessed: Jan-2023. 2023. URL: http://graphics.stanford.edu/data/3Dscanrep 5.

[Suz*85] SUZUKI, SATOSHI et al. "Topological structural analysis of digitized binary images by border following". *Computer vision, graphics, and image processing* 30.1 (1985), 32–46 4.

[TAW*19] TURSUN, OKAN TARHAN, ARABADZHIYSKA-KOLEVA, ELENA, WERNIKOWSKI, MAREK, et al. "Luminance-contrast-aware foveated rendering". *ACM Transactions on Graphics (TOG)* 38.4 (2019), 1–14 2.

[TD22] TURSUN, CARA and DIDYK, PIOTR. "Perceptual Visibility Model for Temporal Contrast Changes in Periphery". *ACM Trans. Graph.* 42.2 (Nov. 2022). ISSN: 0730-0301. DOI: 10.1145/3564241 4–6.

[TEHM96] TSUMURA, NORIMICHI, ENDO, CHIZUKO, HANEISHI, HIDEAKI, and MIYAKE, YOICHI. "Image compression and decompression based on gazing area". *Human Vision and Electronic Imaging*. Vol. 2657. SPIE. 1996, 361–367 2.

[TLTT08] TO, M, LOVELL, P GEORGE, TROSCIANKO, TOM, and TOLHURST, DAVID J. "Summation of perceptual cues in natural visual scenes". *Proceedings of the Royal Society B: Biological Sciences* 275.1649 (2008), 2299–2308 4.

[TRK20] TIWARY, ANKUR, RAMANATHAN, MUTHUGANAPATHY, and KOSINKA, JIRI. "Accelerated Foveated Rendering based on Adaptive Tessellation". English. *Eurographics 2020 - Short Papers*. Ed. by WILKIE, ALEXANDER and BANTERLE, FRANCESCO. The Eurographics Association, 2020. ISBN: 978-3-03868-101-4. DOI: 10.2312/egs.20201003 3.

[TTD22] TARIQ, TAIMOOR, TURSUN, CARA, and DIDYK, PIOTR. "Noise-Based Enhancement for Foveated Rendering". *ACM Trans. Graph.* 41.4 (July 2022). ISSN: 0730-0301. DOI: 10.1145/3528223.3530101 2.

[Uni23] UNITY. *Asset store*. Accessed: Jan-2023. 2023. URL: https://assetstore.unity.com 5.

[VL06] VELTKAMP, REMCO C and LATECKI, LONGIN JAN. "Properties and performance of shape similarity measures". *Data Science and Classification*. Springer, 2006, 47–56 2.

[WP08] WEDEL, MICHEL and PIETERS, RIK. "A review of eye-tracking research in marketing". *Review of marketing research* (2008) 2.