Luminance-Contrast-Aware Foveated Rendering

OKAN TARHAN TURSUN, Max Planck Institute for Informatics
ELENA ARABADZHIYSKA-KOLEVA, Saarland University, MMCI and Max Planck Institute for Informatics
MAREK WERNIKOWSKI, West Pomeranian University of Technology
RADOSŁAW MANTIUK, West Pomeranian University of Technology
HANS-PETER SEIDEL, Max Planck Institute for Informatics
KAROL MYSZKOWSKI, Max Planck Institute for Informatics
PIOTR DIDYK, Università della Svizzera Italiana

Fig. 1. Current foveated rendering techniques (left) use a fixed quality decay for peripheral vision. While this can be a conservative solution, it does not provide a full computational benefit. Our technique (right) performs content-adaptive foveation and relaxes the quality requirements for content for which the sensitivity of the human visual system at large eccentricities degrades faster. Image by pixel2013 / Pixabay.

Current rendering techniques struggle to fulfill quality and power efficiency requirements imposed by new display devices such as virtual reality headsets. A promising solution to overcome these problems is foveated rendering, which exploits gaze information to reduce rendering quality for the peripheral vision where the requirements of the human visual system are significantly lower. Most of the current solutions model the sensitivity as a function of eccentricity, neglecting the fact that it also is strongly influenced by the displayed content. In this work, we propose a new luminance-contrast-aware foveated rendering technique which demonstrates that the computational savings of foveated rendering can be significantly improved if local luminance contrast of the image is analyzed. To this end, we first study the resolution requirements at different eccentricities as a function of luminance patterns. We later use this information to derive a low-cost predictor of the foveated rendering parameters. Its main feature is the ability to predict the parameters using only a low-resolution version of the current frame, even though the prediction holds for high-resolution rendering. This property is essential for the estimation of required quality before the full-resolution image is rendered. We demonstrate that our predictor can efficiently drive the foveated rendering technique and analyze its benefits in a series of user experiments.

CCS Concepts: • **Computing methodologies** → **Perception**; **Rendering**; **Image manipulation**;

Additional Key Words and Phrases: foveated rendering, perception

ACM Reference Format:
Okan Tarhan Tursun, Elena Arabadzhiyska-Koleva, Marek Wernikowski, Radosław Mantiuk, Hans-Peter Seidel, Karol Myszkowski, and Piotr Didyk. 2019. Luminance-Contrast-Aware Foveated Rendering. *ACM Trans. Graph.* 38, 4, Article 98 (July 2019), 14 pages. https://doi.org/10.1145/3306346.3322985

1 INTRODUCTION

New display designs, such as virtual and augmented reality glasses, may revolutionize the way we interact with the virtual and the real worlds. While virtual reality (VR) enables us to experience new, unknown environments which we would not be able to explore otherwise, augmented reality (AR) allows us to enrich reality with digital information. For these technologies to succeed, the visual quality delivered by the new devices has to first meet the capabilities and the requirements of the human visual system (HVS). In par-

Authors' addresses: Okan Tarhan Tursun, Max Planck Institute for Informatics, okan.tursun@mpi-inf.mpg.de; Elena Arabadzhiyska-Koleva, Saarland University, MMCI, Max Planck Institute for Informatics, earabadz@mpi-inf.mpg.de; Marek Wernikowski, West Pomeranian University of Technology, mwernikowski@wi.zut.edu.pl; Radosław Mantiuk, West Pomeranian University of Technology, rmantiuk@wi.zut.edu.pl; Hans-Peter Seidel, Max Planck Institute for Informatics, hpseidel@mpi-inf.mpg.de; Karol Myszkowski, Max Planck Institute for Informatics, karol@mpi-inf.mpg.de; Piotr Didyk, Università della Svizzera Italiana, piotr.didyk@usi.ch.

0730-0301/2019/7-ART98 $15.00
https://doi.org/10.1145/3306346.3322985

Fig. 2. The same foveation exhibits different visibility depending on the underlying texture. In this image, the foveation was optimized such that it is invisible for the photograph (left part). At the same time, however, it can be easily detected on the text texture (right part).

ticular, the new displays have to deliver high spatial and temporal resolution using low-power computational units to maintain the small form factor of the setup. Despite significant improvements in display devices, rendering software, and hardware, computational efficiency and bandwidth are still significant factors limiting visual quality achieved by novel display technologies [Vieri et al. 2018]. Therefore, it remains a significant challenge to develop efficient rendering techniques which match the quality required by the HVS at minimal computational cost.

Great promise for improving rendering quality and power efficiency lies in exploiting human perception [Masia et al. 2013]. The quality perceived by a human subject is not uniform across the visual field, but decreases towards the periphery [Noorlander et al. 1983; Prince and Rogers 1998; Strasburger et al. 2011]. This observation provided a foundation for foveated rendering – gaze-contingent rendering methods that provide the highest quality only for foveal vision and degrade it towards the periphery without visible artifacts [Duchowski and McCormick 1995; Weier et al. 2017]. Due to the rapid development of low-cost eye trackers, foveated rendering will play a key role for new VR devices [Durbin 2017]. Although the benefits of such an approach have been successfully demonstrated for many quality attributes, e.g., spatial resolution [Guenter et al. 2012; Patney et al. 2016; Stengel et al. 2016a], color [Duchowski et al. 2009], and depth [Kellnhofer et al. 2016], we show that these techniques do not fully exploit their potential. In particular, most of the existing techniques propose to degrade the rendering quality as a function of eccentricity, but neglect the fact that the sensitivity of the HVS to image distortions also depends on the underlying content - the effect known as visual masking. A relevant observation for our work is that the visibility of foveation depends on the underlying luminance contrast, i.e., while a given reduction of spatial resolution becomes objectionable in high-contrast regions, it remains unnoticed for low-contrast regions (Figure 2). As we later show in this paper (Section 5), this observation is confirmed by our measurements for different visual eccentricities, which show a significant difference in the tolerable amount of quality degradation depending on the underlying visual content (Figure 7).

In this paper, we exploit the above observation and propose a luminance-contrast-aware foveated rendering strategy. In contrast to previous techniques, our method adjusts the spatial resolution not only according to the eccentricity but also to the underlying luminance information, taking into account the strength of local visual masking. To this end, we propose a new, low-cost predictor that takes a current frame as an input and provides a spatially-varying map of required spatial resolution. The predictor is based on existing models of visual masking, but it is trained for foveated rendering on a new dataset acquired in a psychophysical experiment. We demonstrate that such prediction can be accurate even if the input is a low-resolution frame. This property is critical, as it allows us to predict the required parameters of foveated rendering based on a crude approximation of the new frame. To apply the prediction in foveated rendering, we first render a low-resolution version of a frame to which we apply our predictor. Next, we render the frame according to the predicted quality. We demonstrate that this strategy leads to substantial computational savings without reducing visual quality. The results are validated in a series of user experiments including full foveated rendering systems. The main contributions of this work include:

- an efficient data collection procedure for testing visibility of foveation for a wide field-of-view,
- perceptual experiments investigating the visibility of spatial resolution reduction as a function of eccentricity and underlying luminance signal for complex image patches,
- an efficient prediction of required spatial resolution based on a low-resolution input frame,
- application of the predictor to foveated rendering in desktop- and HMD-based end-to-end systems with eye tracking.

2 RELATED WORK

In this section, we briefly discuss visual models of contrast perception, and their extensions that can handle retinal eccentricities. Since our key application is the reduction of spatial resolution in foveated rendering, we focus on blur perception modeling (Section 2.1). We also discuss previous techniques for gaze-contingent rendering, and emphasize those solutions that account for local image content (Section 2.2).

2.1 Perceptual Background

Contrast perception. Image contrast is one of the most important features for visual perception [Peli 1990]. Contrast detection depends on the spatial frequency of a contrast pattern, and it is characterized by *the contrast sensitivity function (CSF)* [Barten 1999]. The perceived contrast is a non-linear function of contrast magnitude, and the incremental amount of detectable contrast change increases with the contrast magnitude. This effect is often called *self-contrast masking*, and it is modeled using compressive contrast transducer functions [Lubin 1995; Zeng et al. 2001]. The contrast detection threshold also increases with the neighboring contrast of similar spatial frequency [Legge and Foley 1980]. To analyze this *spatial masking* effect, often band-pass filter banks are used first to decompose an image into different frequency *channels* [Lubin 1995; Mantiuk et al. 2011b; Zeng et al. 2001], and then quantify the

amount of masking within each channel separately. Majority of perceptual models that are used in various applications, such as image quality evaluation [Lubin 1995; Mantiuk et al. 2011b], compression [Zeng et al. 2001], and rendering [Bolin and Meyer 1998; Ramasubramanian et al. 1999], consider all the HVS characteristics mentioned above. In this work, we take a similar approach, but we account for the loss of contrast sensitivity in the peripheral vision and aim for a computationally efficient solution that can be used in foveated rendering.

Peripheral vision. Perceptual characteristics of the HVS and, in particular, contrast perception, are not homogeneous over the visual field. This non-homogeneity is often related to the non-uniform distribution of retinal sensory cells. To explain the perceptual difference between foveal and peripheral vision, Curcio and Allen [1990] provide anatomical measurements of ganglion cell densities as a function of retinal eccentricity. In more recent work, Watson [2014] parameterizes this relation with a formula for four different visual quadrants and compares the estimations of cell densities with actual measurements from previous studies. Such a parameterization allows computation of the Nyquist frequency for an arbitrary position in the visual field based on the sampling rate of retinal ganglion cells. However, such anatomical models do not fully explain peripheral sensitivity to visual features, such as contrast. Peli et al. [1991] address this gap and extend the foveal CSF to the peripheral visual field. Although their extension fits well to the previous peripheral contrast sensitivity measurements, it is not a complete model for foveated rendering. In our work, we extend this approach by providing an end-to-end system for the estimation of the required quality of contrast reproduction in complex images across wide field-of-view and using it in the image synthesis task.

Blur sensitivity. The decreased sensitivity to image distortions in peripheral vision motivates foveated rendering techniques (Section 2.2) to save computation time by rendering low-resolution content at larger eccentricities. From the perception point of view, the closest effect extensively studied in the literature is blur perception. For foveal vision, many studies measure detection and discrimination threshold for simple stimuli such as a luminance edge blurred with different Gaussian filters [Watson and Ahumada 2011]. Similar experiments can be used to measure the sensitivity to blur at various eccentricities [Kim et al. 2017; Ronchi and Molesini 1975; Wang and Ciuffreda 2005]. The existing studies reveal a monotonic increase in the threshold values as a function of eccentricity. Unfortunately, simple stimuli used in the above experiments cannot represent the rich statistical variety of complex images. In particular, such threshold values strongly depend on the image content [Sebastian et al. 2015]. In this work, we generalize these findings regarding the blur perception beyond the fovea and investigate the content-dependent blur sensitivity at various retinal eccentricities.

Image metrics. Perceptual experiments studying the sensitivity of the HVS to contrast changes can be used for developing image metrics which are then used to evaluate or drive image synthesis techniques. In this context, Watson and Ahumada [2011] argue that when the reference and blurred images are given as inputs, general models of contrast discrimination can account for blur perception for simple stimuli in the fovea. Their model works by summing the energy over a restricted local extent and uses the CSF as well as the spatial contrast masking effects. Sebastian et al. [2015] employ a similar generic model to predict their data for complex images, while Bradley et al. [2014] additionally consider local luminance adaptation to account for near eccentricity (up to 10°). The closest to our efforts is the work by Swafford et al. [2016] which extends the advanced visible difference predictor HDR-VDP2 [Mantiuk et al. 2011b] to handle arbitrary eccentricities by employing a cortex magnification factor to suppress the original CSF. The authors attempt to train their metric based on data obtained for three applications of foveated rendering, but they cannot find a single set of parameters that would fit the metric prediction to the data. In this work, we draw from these findings but aim for a computationally efficient solution which accounts for complex images and can be used to drive foveated rendering techniques.

2.2 Foveated Rendering

Traditional techniques. Gaze-contingent rendering has many potential applications focused on the improvement of viewing experience and reduction of the computation costs (refer to [Weier et al. 2017] for a recent survey). Gaze-driven solutions contribute to the improvement of tone mapping [Jacobs et al. 2015], depth perception [Kellnhofer et al. 2016] and viewing comfort in stereoscopic displays [Duchowski et al. 2014]. Computational depth-of-field effects partially compensate for the lack of proper eye accommodation in standard displays [Mantiuk et al. 2011a; Mauderer et al. 2014], while for displays with accommodative cues, proper alignment of multi-focal images can be achieved [Mercier et al. 2017] or laser beams can be guided by pupil tracking [Jang et al. 2017]. The computation performance may be improved by reducing the level of detail [Duchowski et al. 2009; Reddy 2001], or spatial image resolution [Guenter et al. 2012; Patney et al. 2016; Stengel et al. 2016b; Sun et al. 2017; Swafford et al. 2016; Vaidyanathan et al. 2014] towards the periphery, which is particularly relevant for this work.

Content-dependent techniques. Image content analysis to improve quality and efficiency in foveated rendering has been considered to a relatively limited extent. Patney et al. [2016] use contrast enhancement to help recover peripheral details that are resolvable by the eye but degraded by filtering that is used for image reconstruction from sparse samples. Stengel et al. [2016b] make use of information available from the geometry pass, such as depth, normal, and texture properties, to derive local information on silhouettes, object saliency, and specular highlights. The combined features along with visual acuity fall-off with the eccentricity and luminance adaptation state (based on the previous frame) allow for sparse sampling of costly shading. As luminance information is not available before shading for the current frame, contrast sensitivity and masking cannot be easily considered. Sun et al. [2017] propose a foveated 4D light field rendering with importance sampling that accounts for focus differences between scene objects, which are determined by the object depth and the eye accommodation status at the fixation point. This leads to the reduction of computation costs for optically blurred scene regions, which requires displays that can trigger the eye accommodation. Our content-dependent processing does not

account for depth differences but rather refers to contrast and blur perception on standard displays. We are inspired by previous work that refers to contrast perception to improve the rendering performance for foveal vision [Bolin and Meyer 1998; Ramasubramanian et al. 1999], but we consider a foveated rendering setup that imposes additional constraints on the efficiency of our visual model.

3 OVERVIEW AND MOTIVATION

Our approach relies on a new computational model for luminance contrast (Section 4), which estimates the maximum spatial resolution loss that can be introduced to an image without visible artifacts. It is based on underlying content and eccentricity which are important in the context of foveated rendering. The model relies on characteristics of the HVS such as the peripheral contrast sensitivity and a transducer model for contrast perception. We calibrate the model prediction using our new experimental data (Section 5). Our technique relies on two critical observations described below.

Hoffman et al. [2018] and Albert et al. [2017] demonstrate that temporarily-stable low-resolution rendering is perceptually equivalent to a Gaussian-blurred high-resolution rendering. This motivates our technique to model the resolution reduction using a Gaussian low-pass filter. Consequently, our model uses a standard deviation (σ_s) of the Gaussian filter to express the maximum acceptable resolution reduction. The σ_s value can be later translated into the rendering resolution for given content and used to drive rendering resolution adaptively during real-time rendering (Figure 3). Thanks to the above assumption, we derive our model as a closed-form expression, which enables an efficient implementation.

The decision about the optimal rendering resolution would be made best based on full information about the content, i.e., complete contrast information across different spatial frequencies. However, this would require a full-resolution rendering in the first place, and therefore, it is not a feasible solution in foveated rendering. Due to this paradoxical nature of the problem, we first design and test our predictor using high-resolution inputs. Later, we show that it is possible to re-train the model such that it provides the prediction based on a low-resolution rendering. In the latter case, undersampled high-frequency features are still present in a form of aliasing which conveys to our metric information on local contrast localization.

4 COMPUTATIONAL MODEL

In this section, we derive a computational model that estimates the maximum resolution reduction that remains undetectable by an observer. The derivation operates on local patches of high-resolution image and computes a standard deviation of a Gaussian low-pass filter which models the resolution degradation. We derive the model in two steps. First, we express the luminance contrast of the patch in perceptual units (Section 4.1). Based on this measure, we derive a formula for computing the standard deviation σ_s (Section 4.2).

4.1 Perceptual Contrast Measure

We express the perceived contrast of a single image patch as a function of spatial frequency and eccentricity. The function accounts for contrast sensitivity of the human visual system as well as visual masking (Figure 4). The model, as described here, contains several

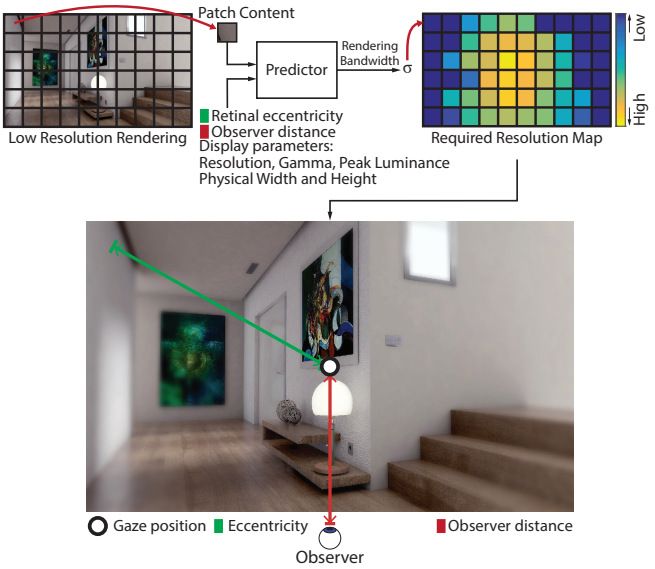

Fig. 3. Overview of our method. Our predictor takes patches, retinal eccentricity, observer distance and display parameters such as the resolution, gamma, peak luminance, physical width and height as inputs and predicts the required spatial rendering bandwidth expressed as the standard deviation of a low-pass Gaussian filter. The map is generated using our method and the output is enhanced for visibility. Image by Pxhere.

free parameters which we optimize based on experimental data (Section 5).

Luminance contrast. The process starts with the conversion of the intensity of every pixel p to an absolute luminance value $L(p)$. Next, we compute band-limited contrast similarly to [Lubin 1995; Ramasubramanian et al. 1999]. To this end, we first perform a Laplacian pyramid decomposition [Burt and Adelson 1983] which provides band-limited luminance difference $\Delta L(f,p)$. Then, following [Peli 1990], we use the decomposition to compute the luminance contrast pyramid as:

$$C(f,p) = \frac{\Delta L(f,p)}{L_a(f,p) + \epsilon}, \tag{1}$$

where f is the spatial frequency in *cpd* units (cycles-per-visual-degree) and ϵ is a small number to prevent mathematical singularities in the regions with low luminance. The average luminance $L_a(f,p)$ in the denominator is provided by the corresponding point in the Gaussian pyramid two levels down in resolution, which is upsampled by a factor of four using a linear interpolation.

Contrast sensitivity and retinal eccentricity. To obtain information about the magnitude of perceived contrast, we normalize the values in the pyramid using eccentricity-dependent contrast sensitivity function (CSF). This gives us luminance contrast C_n expressed as a multiple of detection threshold:

$$C_n(f,p) = C(f,p)\, S_{\text{CSF}}\big(f, \theta(p), L_a(f,p)\big), \tag{2}$$

where $\theta(p)$ is the retinal eccentricity of pixel p expressed in visual degrees, and $L_a(f,p)$ models the adaptation luminance. Here, we base the contrast sensitivity function S_{CSF} on [Peli et al. 1991] where the

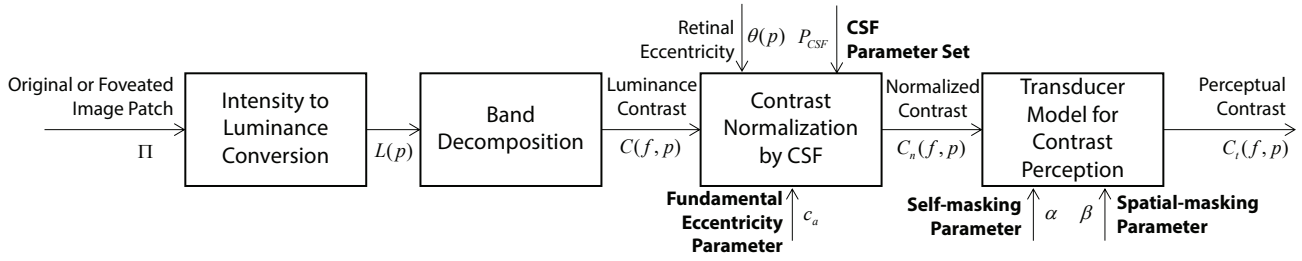

Fig. 4. This figure shows a flowchart of our model for computing the perceptual contrast measure. The input parameters which are optimized during the calibration are shown in bold.

standard contrast sensitivity function S'_{CSF} for the fovea is attenuated according to the eccentricity:

$$S_{\text{CSF}}(f,\theta,L_a) = \frac{1}{\exp(c_a\theta f)}a(L_a)S'_{\text{CSF}}(f). \tag{3}$$

In the above equation, c_a is the fundamental eccentricity parameter that models the rate of HVS acuity loss in the peripheral visual field, f is the spatial frequency, and $a(L_a) = (1 + 0.7/L_a)^{-0.2}$ represents the effect of adaptation luminance L_a on the peak sensitivity [Barten 1989]. After initial attempts of using existing CSF definitions such as [Barten 1989; Mannos and Sakrison 1974] for S'_{CSF}, we opted for a custom solution. We define the CSF at four frequency bands centered at 4, 8, 16 and 32 cpds with values denoted by s_4, s_8, s_{16} and s_{32} (parameters of our model). The sensitivities for the intermediate frequencies are obtained using cubic Hermite spline interpolation in the log-sensitivity and log-frequency domain. We found that this solution provides a more accurate prediction of our model than using standard CSF functions. We attribute this behavior to a broad-band characteristic of a Laplacian pyramid[1], which is better handled by a custom definition which accounts for broad-band stimuli in contrast to standard CSF which is derived for a single luminance frequency stimuli.

Visual masking. In the final step of measuring the perceived luminance contrast, we incorporate the effect of the visual masking. To this end, we use the transducer model of Zeng et al. [2000] on the normalized contrast C_n, and expressed the final value of perceived luminance contrast as:

$$C_t(f,p) = \frac{\text{sign}(C_n(f,p)) \cdot |C_n(f,p)|^{\alpha}}{1 + \frac{1}{|N|}\sum_{q\in N(p)} |C_n(f,q)|^{\beta}}, \tag{4}$$

Here, the numerator models the self-masking effects, while the denominator models the spatial-masking effects from the 5×5 neighborhood $N(p)$ in the same band. $\alpha, \beta \in [0, 1]$ are parameters of our model, which control the masking modeling.

4.2 Estimation of Resolution Reduction

Our goal is to estimate per-patch maximal resolution reduction that would remain unnoticeable by an observer. Since we model the resolution reduction using a Gaussian low-pass filter, we are seeking a maximum standard deviation for the Gaussian filter such that the difference between the original patch and its filtered version will be imperceptible. Using our perceived contrast definition from the previous section (Equation 4), we can formalize this problem as:

$$\begin{aligned} &\text{maximize} \quad \sigma_s, \\ &\text{subject to} \quad \forall_{p\in\Pi,f}\ C_t(p,f) - C'_t(p,f) \le 1, \end{aligned} \tag{5}$$

where $C_t(p,f)$ is the contrast of the original patch Π. In this and the following equations, we use C' notation for all the contrast measures related to the patch Π convolved with G_{σ_s}, a Gaussian function with standard deviation equal to σ_s. Consequently, $C'_t(p,f)$ is the perceived contrast measure of the original patch which is pre-filtered using G_{σ_s}. The constraint in this formulation guarantees that the difference between the two patches will be below the visibility threshold. Due to the complex nature of the contrast definition and the spatial dependencies between contrast values for neighboring regions, the above optimization does not have a direct solution and requires an iterative optimization for the entire image. This would be prohibitively expensive in the context of foveated rendering. Therefore, in this section, we demonstrate how this formulation can be simplified leading to a closed-form solution for σ_s.

Let us first consider estimating σ_s for one pixel p and single spatial frequency f. If the patch Π is convolved with G_{σ_s}, the values in the Laplacian frequency decomposition will be attenuated according to the frequency response of the filter. More precisely, the frequency response of G_{σ_s} for frequency f will be given by:

$$\hat{G}_{\sigma_s}(f) = \frac{C'(f,p)}{C(f,p)}. \tag{6}$$

On the other hand, we know that the frequency response of a Gaussian filter G_{σ_s} is also a Gaussian:

$$\hat{G}_{\sigma_s}(f) = \exp\left(-f^2 / \left(2\sigma_f^2\right)\right), \tag{7}$$

where σ_f is the standard deviation in the frequency domain, and it is defined as $\sigma_f = (2\pi\sigma_s)^{-1}$. By combining Equations 6 and 7, one can show that σ_f can be expressed as:

$$\sigma_f = \frac{f}{\sqrt{-2\ln\left(\frac{C'(f,p)}{C(f,p)}\right)}} = \frac{f}{\sqrt{-2\ln\left(\frac{C'_n(f,p)}{C_n(f,p)}\right)}}, \tag{8}$$

where the last transition is a direct consequence of the Equation 2. In the above equation, $C_n(f,p)$ can directly be computed from the input patch. So the only unknown, besides σ_f which we need to

[1]Each band of a Laplacian pyramid contains a broad frequency spectrum, and in particular, the highest frequency band contains a significant portion of medium frequencies.

compute, is $C'_n(f,p)$. To obtain its value, we will use the contrast loss constraint from Equation 5.

We can assume that when σ_s increases, and so does the difference between $C'_t(f,p)$ and $C_t(f,p)$. Thus, when σ_s is a solution to our problem, the following equality holds: $C'_t(f,p) - C_t(f,p) = 1$. We can directly express perceived contrast in this equality using Equation 4, and obtain:

$$\frac{\text{sign}\big(C_n(f,p)\big)\cdot\big|C_n(f,p)\big|^{\alpha}}{1+\frac{1}{|N|}\sum\limits_{q\in N(p)}\big|C_n(f,q)\big|^{\beta}} - \frac{\text{sign}\big(C'_n(f,p)\big)\cdot\big|C'_n(f,p)\big|^{\alpha}}{1+\frac{1}{|N|}\sum\limits_{q\in N(p)}\big|C'_n(f,q)\big|^{\beta}} = 1. \tag{9}$$

It becomes clear, that $C'_n(f,p)$ cannot be computed directly from this equation due to the visual spatial masking term in the denominator of the first component. Therefore, we make one more assumption. We assume that the spatial masking for the path convolved with G_{σ_s} can be approximated by the spatial masking in the original patch. Assuming additionally that the sign of the contrast does not change during the filtering, the above equation can be simplified to:

$$\frac{\text{sign}\big(C_n(f,p)\big)\cdot\Big(\big|C_n(f,p)\big|^{\alpha} - \big|C'_n(f,p)\big|^{\alpha}\Big)}{1+\frac{1}{|N|}\sum\limits_{q\in N(p)}\big|C_n(f,q)\big|^{\beta}} = 1. \tag{10}$$

From the above equation, $C'_n(f,p)$ can directly be derived as:

$$C'_n(f,p) = \left|\big|C_n(f,p)\big|^{\alpha} - \left(1+\frac{1}{|N|}\sum_{q\in N(p)}\big|C_n(f,q)\big|^{\beta}\right)\right|^{1/\alpha}. \tag{11}$$

Please note that we omit the sign of the contrast since we are interested only in its magnitude. The above definition of $C'_n(f,p)$ and Equation 8 provide a closed-form expression for computing optimal σ_f for a particular pixel p and spatial frequency f.

Now, we could simply use the relation $\sigma_f = (2\pi\sigma_s)^{-1}$ to convert σ_f to the primary domain. Before doing this, we first compute σ_f for entire patch, which is critical for our calibration procedure (Section 5). To this end, we first combine σ_f estimation for pixel patch by taking the maximum value across all frequency levels. This allows us to make our method conservative and not overestimate the acceptable resolution reduction. Note that, larger σ_f corresponds to smaller blur, and therefore, smaller acceptable resolution reduction. Next, we combine the obtained values across the entire patch using a smooth maximum function:

$$\hat{\sigma}_f = \left(\sum_{p\in\Pi}\sigma_f(p)\cdot\exp\Big(\omega\cdot\sigma_f(p)\Big)\right)\Big/\left(\sum_{p\in\Pi}\exp\Big(\omega\cdot\sigma_f(p)\Big)\right), \tag{12}$$

where $\omega\in[0,\infty)$ is a parameter which controls the behavior of the function ranging from computing average as $\omega\to 0$ and maximum as $\omega\to\infty$. When experimenting with optimizing parameters of our model, we found that the smooth maximum performs better than simply taking maximum value. Finally, σ_s for the entire patch is computed as:

$$\hat{\sigma_s} = \frac{1}{2\pi\hat{\sigma}_f}. \tag{13}$$

5 CALIBRATION

Our model is defined using several free parameters: self-masking parameter (α), spatial-masking parameter (β), CSF parameters (s_4, s_8, s_{16}, s_{32}), fundamental eccentricity (c_a), and smooth max parameter (ω). In this section, we present a calibration procedure and perceptual experiments that are used to collect necessary user data.

Training our model requires a set of patch pairs consisting of a high-resolution patch as well as its low-resolution version for which the quality degradation is not detectable. One way of collecting such data is measuring maximum and undetectable resolution reduction for individual patches and eccentricities. However, such procedure limits each trial to a single eccentricity and patch. As a result, it requires long sessions to collect data. Instead, we propose to gather the data in a more efficient way. We tile one patch into an image covering the entire screen and estimate the optimal, unnoticeable foveation. This allows us to derive necessary information for a whole range of eccentricities.

We define foveated rendering using two parameters. The first one is the radius r of the foveal region where the visual content is rendered in the highest resolution. The second parameter is the rate k at which the resolution is reduced towards the periphery. Consequently, the resolution reduction modeled by using a standard deviation of a Gaussian filter can be expressed as:

$$\sigma_s(\theta) = \begin{cases} 0, & \text{if } \theta < r, \\ k\cdot(\theta - r), & \text{if } \theta \geq r, \end{cases} \tag{14}$$

where θ is the retinal eccentricity of a particular point on the screen. Different combinations of r and k affect the tradeoff between quality reduction and rendering efficiency. The goal of our experiment is to measure the visibility of foveation for different parameters and image patches to determine the strongest one which remains unnoticeable.

The experimental setup. Our system consists of a Tobii TX300 Eye Tracker, a chin-rest to keep the observation point relatively stable during the experiments, and two displays. The first one is a 27" ASUS PG278Q display with 2560×1440 resolution spanning a visual field of $48.3^\circ\times28.3^\circ$ from a viewing distance of 66.5 *cm*. The second display is a 32" Dell UP3216Q with 3840×2160 resolution spanning a visual field of $52.3^\circ\times30.9^\circ$ from a viewing distance of 71 *cm*. The peak luminances of the displays are measured as 214.6 cd/m^2 and 199.2 cd/m^2 whereas the peak resolutions produced at the center are 24.9 *cpd* and 34.1 *cpd* for the first and the second displays, respectively.

The stimuli. Our dataset consists of 36 patches selected from natural and synthetic images with different characteristics (see Figure 5). We picked the first 18 patches randomly from a large set of natural images [Cimpoi et al. 2014]. To improve the diversity of our dataset, the remaining 18 patches are picked from a large set of 5640 images by maximizing the dissimilarity between patches (Appendix A). For the final stimuli, we fill the display frame by tiling an input patch. We avoid introducing high-frequency components on the transitions between tiles by mirroring them about the vertical axis, when tiling in the horizontal direction, and about the horizontal axis, when tiling in the vertical direction (see Figure 6).

The foveated version of the stimuli are prepared by filtering using a Gaussian kernel with standard deviation given by Equation 14. We used 9 different combinations of r and k ($r \in \{4, 7, 11\}$ and $k \in \{0.0017, 0.0035, 0.0052\}$).

The procedure. We use a 2AFC procedure, where the alternatives are foveated and non-foveated versions of the same stimuli. Participants were asked to choose the image which did not contain foveation. In order to get the participants familiar with the experiment interface and controls, we included a training stage at the beginning of the experiment, where the concept of foveation was explained using an exaggerated example. The stimuli were prepared with the assumption that the gaze position is located at the center of the display. The stimulus was hidden when the gaze position deviated from the center. Different alternatives were indicated by randomly assigned letters (A and B) at the center of the screen. The observers were able to switch between two alternatives freely, and they were shown a uniform gray screen in-between. A total of 8 participants with normal or corrected-to-normal vision participated in our experiment, and they made a total of 324 comparisons in six sessions for 36 patches. Each comparison was repeated 10 times by each participant and the average time required for a participant to complete a single session was 40 minutes.

Results. From the results of the above experiment, we want for each patch to compute an optimal σ_s as a function of eccentricity. To this end, we first compute for each patch i the probability of detecting foveation given by triplet (r, k, θ) as:

$$P(det|r,k,\theta) = \frac{1}{N}\sum_{n=1}^{N} a_n(r,k,\theta), \tag{15}$$

$$a_n(r,k,\theta) = \begin{cases} 1, \text{if non-foveated stimulus is chosen,} \\ 0, \text{otherwise.} \end{cases} \tag{16}$$

where N is the number of comparisons by each participant. If $P(det|r,k,\theta) < 0.75$, we labeled this combination of (r,k,θ) as undetectable. We then for each eccentricity take the maximum value of σ_s across all (r,k,θ) marked as undetected. This defines per-patch and per-observer optimal $\sigma_s(\theta)$. As the last step, we average $\sigma_s(\theta)$ values across the participants to obtain the ground truth $\sigma_s^{(i)}(\theta)$ for patch i. The same procedure is repeated for all patches. The resulting $\sigma_s^{(i)}$ functions are shown in Figure 7. The range marked by the whiskers indicates to what extent the acceptable blur depends on the underlying patch for a particular eccentricity. The significant differences between sigma values for different patches (see insets) are the central insight we use in our work. Please refer to the plots in the supplemental materials, which show more in-depth analysis of measured $\sigma_s^{(i)}$ values.

In our implementation, we choose detection threshold, 0.75, as the middle value between the success rate associated with random guessing ($P(det|r,k,\theta) = 0.50$ with two alternatives) and the probability of a guaranteed detection ($P(det|r,k,\theta) = 1.00$). This value is commonly used in previous perceptual studies to estimate "barely" visible differences [Lubin 1995]. The threshold can be adjusted based on the application requirements, and the ground truth associated with a different threshold probability can be easily computed from existing data without repeating the perceptual experiment.

Optimization. Finally, to find the parameters of our model, we use a hybrid optimization approach. In the first stage, we use Adaptive Simulated Annealing (ASA) [Aguiar e Oliveira Junior et al. 2012] to optimize for predictor parameters. In the second stage, we run a gradient-based minimization to fine-tune the result of ASA. This hybrid optimization scheme helps avoiding local optima. The following weighted Mean Absolute Error (MAE) is minimized during the optimization:

$$E = \min_{\mathbb{S}} \frac{1}{36}\sum_{i=1}^{36}\sum_{\theta=4^\circ}^{30^\circ} w_1(\theta)\left| w_2\left(\hat{\sigma}_s^{(i)}(\theta) - \sigma_s^{(i)}(\theta)\right)\right|, \tag{17}$$

where $\mathbb{S} = \{\alpha, \beta, c_a, s_4, s_8, s_{16}, s_{32}, \omega\}$ is the set of model parameters, $\sigma_s^{(i)}(\theta)$ is the ground truth for patch i from our experiment and $\hat{\sigma}_s^{(i)}(\theta)$ is the result of our predictor for the eccentricity θ. w_1 and w_2 are weighting functions defined as:

$$w_1(\theta) = \begin{cases} 2, & \text{if } \theta < 10, \\ 1, & \text{otherwise} \end{cases} \tag{18}$$

$$w_2(x) = \begin{cases} 8x, & \text{if } x > 0, \\ x, & \text{otherwise.} \end{cases} \tag{19}$$

The first function, w_1 puts more emphasis on the error measured in the parafoveal region where HVS has a higher sensitivity. On the other hand, the second function, w_2, penalizes underestimation of spatial bandwidth with a larger weight, because underestimation is less desirable than overestimation due to potential visual artifacts.

We check the generalized performance by performing 6-fold cross-validation. Optimal parameters and errors measured at each fold are depicted in the supplementary materials. We observe that the test errors are close to training errors and optimal parameter values are stable among different cross validation folds. As expected, higher training and testing errors are observed when the predictor is calibrated on the inputs with reduced resolution due to the loss of information on high-frequency bands. But we can still assume a reasonable approximation by the predictor due to the small difference in MAE (0.503 compared to 0.554). The optimal parameter values that we are using in our validation are obtained by calibrating our predictor using the whole dataset. These values are given in Table 1.

As explained in Section 3, a method for predicting the acceptable resolution degradation operating on a full-resolution image is not very useful in the context of foveated rendering. Therefore, we take advantage of our parametric design of the model and train it on low-resolution images, which can be provided by foveated rendering as a cheap initial approximation of the frame. Consequently, we calibrate and run our model on inputs which are downsampled by a factor of 1/4 (corresponding to 1/16 of the target area). The downsampling routine employed during the calibration is nearest-neighbor downsampling and it does not involve low-pass filtering or interpolation. This is equivalent to actual low-resolution rendering, including spatial aliasing effects that may arise during rendering. This way, our model is able to utilize cues which appear in the form of aliasing in the presence of higher-frequency content. When actual rendering takes place, we render true low resolution. This gives the optimal performance for the actual rendering applications.

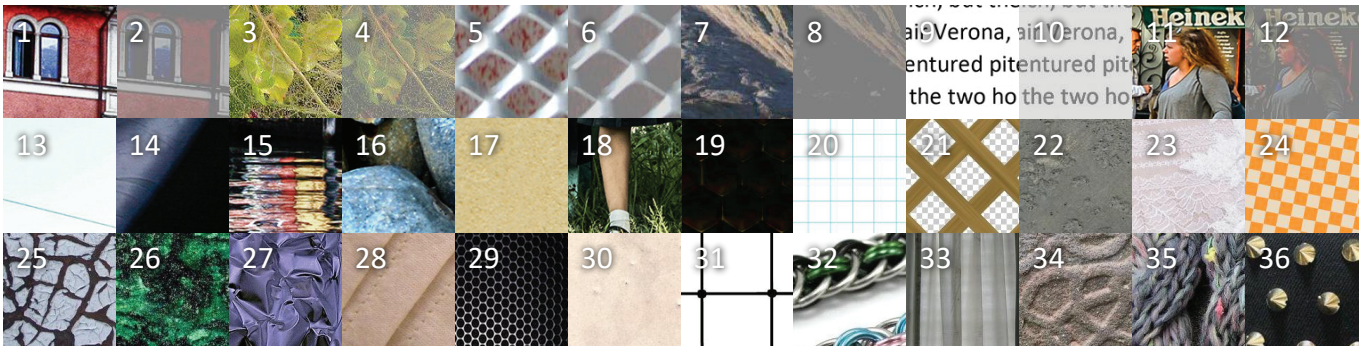

Fig. 5. Our dataset for the calibration of our predictor. We include patches with different luminance and texture patterns from a dataset of natural and synthetic images [Cimpoi et al. 2014]. Note that patches 1-12 contain reduced-contrast versions of the same content to cover a wider range of contrasts during the calibration phase.

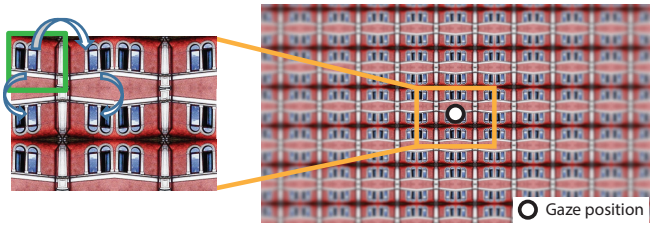

Fig. 6. A sample stimulus used for data collection and calibration. The zoomed region shows how the input patch is tiled prior to Gaussian filtering. For different values of foveal region radius r and rate of quality drop-off k, the participants are asked to compare foveated (shown here) and non-foveated (without Gaussian blur) versions in a 2AFC experiment.

Table 1. Best parameter values obtained after calibration. The input patches are downscaled by a factor of 1/4. Loss is the training error computed using Equation 17. In addition to the loss function, which is a weighted mean absolute error, we also provide the standard unweighted mean absolute error (MAE) for evaluation.

α	β	c_a	$\log_{10}(s_4)$	$\log_{10}(s_8)$
0.555	0.135	0.040	5.290	6.226
$\log_{10}(s_{16})$	$\log_{10}(s_{32})$	ω	Loss	MAE
3.404	4.011	1.919	0.706	0.554

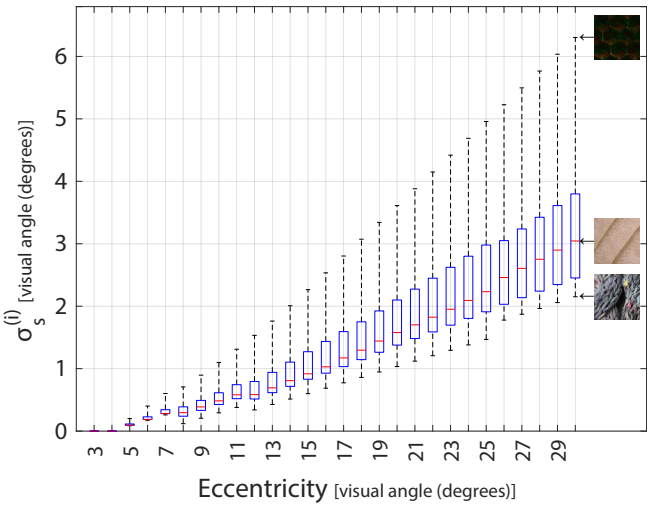

Fig. 7. Box plot of ground truth $\sigma_s^{(i)}$ obtained from our experiment. This plot shows how content influences the tolerable amount of foveation with respect to eccentricity. Red lines represent the median while boxes show the range between 25th and 75th percentile of the data. Whiskers extend to the whole range. The patches which have the minimum, the median and the maximum $\sigma_s^{(i)}$ are shown on the plot for 30° eccentricity.

6 IMPLEMENTATION

We implemented our predictor on desktop and HMD platforms using different rendering API and game engines. The model is implemented in OpenGL Shading Language (GLSL) [2013] and Unity3D [2018] shading languages. The rendering frameworks for our perceptual validation experiments use Unity3D game engine and NVIDIA Variable Rate Shading (VRS) API [2018] on OpenGL.

6.1 Predictor

For our validation experiments, we implemented our method in C++ using the OpenGL Shading Language (GLSL). We adapted our implementation of the model described in Section 4 to fully benefit from the optimizations in GLSL. For example, we implement local operations defined on patches and pixels (such as those given in Equations 1-13) in a way that allows the graphics card to process the whole frame in parallel. Similarly, the decomposition into different frequency bands is achieved using mipmaps to maximize the parallelism, where higher levels represent lower frequencies. σ_f estimation from different frequencies are combined by performing a level-of-detail texture lookup in the pyramid for efficiency. On the other hand, the smooth max function is executed by computing the mipmap of the optimal standard deviation and taking the level in the pyramid which corresponds to the maximum achievable level for a single patch. In all operations, we preserve the patch-pixel correspondence to maintain locality information. We used different patch sizes for two displays during calibration; namely, 128×128

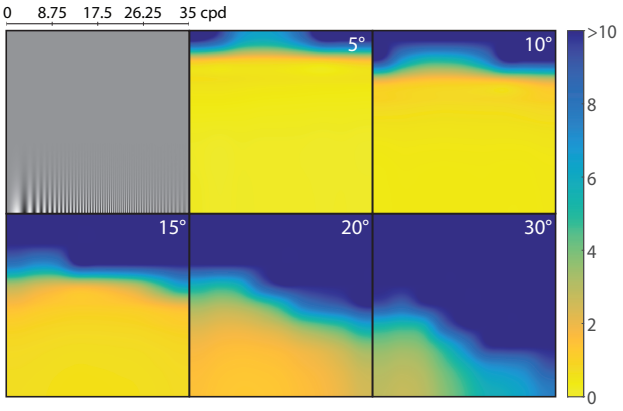

Fig. 8. We used a Campbell-Robson chart (top left) as a test input for our predictor. The predictions of σ_s from our model are given for different visual eccentricities. The eccentricities are indicated at the top-right corner of each map. Our model successfully predicts a higher σ_s (corresponding to a lower rendering resolution) for the spatial frequencies that are imperceptible by the HVS as the visual eccentricity increases and the contrast declines. (Please note that the Campbell-Robson chart is prone to aliasing when viewed or printed in low resolution. Please refer to the electronic copy of this document for a correct illustration.)

for the Asus display (2560×1440) and 192×192 for the Dell display (3840×2160). For rendering, 128×128 was used (corresponds to 32×32 effective patch size for 1/4 downsampled inputs). The choice of patch size is mainly dependent on the number of bands required in Band Decomposition step. Using smaller patches bring a limitation on the pyramid decomposition while using larger patches make the predictions less sensitive to local changes in content. A patch size of 128×128 provides a good balance between these two.

Performance. Table 2 shows the performance benchmark of the predictor implementation on an NVIDIA GeForce GTX 2080Ti graphics card. Since our setup consists of two different displays we provide the measurements for their respective resolutions. For comparison, we also include the time measurements using full resolution inputs.

Table 2. Running times of our implementation. Our predictor is calibrated and validated using 1/4× downsampled inputs (1/16× of the area) in a series of subjective experiments. Here, we show the computational savings obtained by our approach with respect to the predictions from full-resolution inputs. Please note that these values do not include rendering costs.

Input Size	2560×1440	3840×2160
Downsampled (1/4×)	0.7 ms	1.2 ms
Full resolution	3.0 ms	5.9 ms

6.2 Complex Shading

Unity3D and Variable Rate Shading implementations perform two rendering passes. The first pass is the low-resolution off-screen rendering of the scene (1/16 of the full frame area). In the second pass, the actual rendering takes place where the foveation is either simulated by the application of a low-pass Gaussian filter to full-resolution frame (Unity3D) or real foveation is implemented by adjusting the sampling rate of local shading blocks (Variable Rate Shading). Both approaches employ an implementation of our predictor in the underlying shader language, which guides the selection of spatially varying rendering bandwidth.

Fig. 9. Preview of the scenes used for evaluating the performance of different foveated rendering systems.
Sponza model by McGuire CGA [2017], Water and Fog shaders by P_Malin / Shadertoy and SebH / Shadertoy, respectively.

The use of Gaussian filter for simulated foveation in the Unity3D implementation provides a good temporal stability, and it was demonstrated to be visually similar to an anti-aliasing which avoids spurious artifacts resulting from undersampling [Albert et al. 2017; Hoffman et al. 2018]. By using this strategy, we make sure that the decisions of participants are primarily based on the visual evaluation of foveation quality throughout our experiments and isolated from other external factors such as the quality of a particular temporal antialiasing routine that might be employed in the pipeline.

The Variable Rate Shading (VRS) implementation shares the same routines with the GLSL implementation of our predictor. The foveation is performed via VRS API calls to locally customize shading rates for 16×16 blocks within the frame. The spatial rendering bandwith, originally computed as σ_s by our predictor, is converted to sampling rate by defining a cutoff frequency for the Gaussian filter. For our applications, we chose the cutoff frequency corresponding to $2\hat{\sigma}_s$, which is approximately 13.5% of the Gaussian filter's maximum. With this cutoff point choice, the sampling rate required in one dimension for rendering a patch is defined by Nyquist rate as:

$$SR = \frac{1}{4\hat{\sigma}_s}. \tag{20}$$

Current VRS API does not offer a selection of arbitrary shading rates for 16×16 block units, and only the following rates are available: 1/1, 1/2, 1/4, 1/8 or 1/16. Therefore, in our implementation, we round the prediction to the closest available rate for each of 16×16 pixel block. We further exclude 1/2 and 1/8 rates since they provide a non-uniform resolution degradation in the horizontal and vertical direction, which is not supported by our method. Even though our predictor is able to predict shading rates coarser than 1/16, we do not take advantage of this in our tests.

Performance. We used the NVIDIA VRS to compare the performance of different foveation strategies on three different scenes (Figure 9) with shaders of different complexity. They include a simple Phong shading (Sponza) [McGuire 2017] and more complex ones simulating reflections (Water) and volumetric effects (Fog). The renderings were inspired by shaders posted on Shadertoy.com [2,3], but they were adapted to be appropriate for real-time rendering

[2] https://www.shadertoy.com/view/Xl2XRW
[3] https://www.shadertoy.com/view/XlBSRz

system. Table 3 presents the timings for rendering one frame using different strategies in 2560×1440 resolution. As expected, our technique offers better performance than standard foveation when shader complexity increases. For the very simple Phong shader, when the geometry complexity dictates the rendering performance, our technique cannot provide favorable results.

Table 3. The table presents the comparison between the performance of full-resolution rendering, standard foveation, and our method. The times include all the computations required for rendering one frame. In the parenthesis, we provide a speed-up with respect to the full-resolution rendering.

Scene	Full-resolution	Standard foveation	Our foveation
Sponza	2.6 ms	2.3 ms (1.1 x)	3.0 ms (0.9 x)
Water	9.5 ms	5.3 ms (1.8 x)	4.3 ms (2.2 x)
Fog	22.9 ms	13.9 ms (1.6 x)	5.5 ms (4.2 x)

7 VALIDATION

In this section, we first show the results from rendering bandwidth predictions from our method for different inputs. Then we present our results from two subjective experiments. In the first experiment, the participants compare our method with non-foveated rendering. In the second one, they compare our locally adaptive content-aware foveation method with a globally adaptive foveation strategy.

7.1 Visual Evaluation

In Figure 10, outputs of our predictor for 15 different inputs are shown. The sample inputs and corresponding outputs are grouped according to the platform and framework. Overall, we observe that the model successfully adapts the rendering bandwidth to the content, while taking the peripheral sensitivity loss of the HVS into account. In inputs G1 and G2, we mostly see the effect of defocus. For these inputs, our method suggests a higher rendering resolution (represented by a lower $\hat{\sigma}_s$ prediction) on the objects which are in focus. In inputs G3, G5, U3 and U4, we observe that the majority of rendering budget is allocated to the buildings, which contain a large amount of detail, and a much lower amount of the bandwidth is allocated to the mostly uniform regions in the sky. It is possible to see how the heatmap adapts to the silhouette of the street lamp in input G3, which contains large amount of details on a uniform background. In inputs V1 and V2, we observe that a lower rendering budget is assigned to the regions with low luminance levels, where HVS sensitivity to contrast is reduced. Figure 10 also provides a comparison to a standard foveated rendering in terms of fraction of shaded pixels by our technique when compared to the standard foveated rendering (number in parenthesis). Here, both techniques provide foveation which remains unnoticed. For the standard technique the parameters are fixed across all stimuli.

7.2 Foveated vs. Non-Foveated Rendering

The ground truth that we use for calibration corresponds to a perceived contrast loss with a detection probability of 0.75 (1 JND). If the calibration is successful, the contrast loss in the outputs of our method should be detected approximately with this probability. In order to validate this behavior, we perform a 2AFC subjective experiment, where the participants are asked to compare the results of foveated rendering based on our method and non-foveated rendering. The participants performed comparisons by freely toggling between the two results shown on the display and selecting the result which contains the least amount of blur. We conducted this perceptual experiment using the images given in Figure 10 as the experiment stimuli with GLSL, Unity3D and VRS implementations. Unity3D implementation is tested on both desktop display (ASUS with a Tobii eye tracker) and Head-Mounted Display (HTC Vive with an SMI eye tracker) platforms. During the experiments, we consider different multipliers ($0.5 - 5.0$) to rescale the predicted maps. The main motivation behind testing different scaling factors was to validate whether our model precisely predicts the foveation that is close to the just-detectable one.

In order to avoid bias, the stimuli are shown in randomized order and the participants are unaware of the multiplier value of each stimuli during the experiment. To minimize the training effects and to get the participants familiar with the experiment interface, we included a warmup phase at the beginning of the experiment on a different scene. To test the influence of different hardware on the performance of the model we evaluate it using different platforms. The experiments performed on GLSL and VRS implementations are rendered assuming the gaze position at the center of the display. The actual gaze position is monitored with the eye tracker, and in the presence of significant deviations from the center, the stimuli are hidden by displaying a gray screen. On VRS implementation, the participants were still able to rotate the camera using the mouse. On the other hand, HMD and desktop experiments using Unity3D did not impose such constraints and the scene was rendered using the actual gaze position during the experiment.

In total, 12, 14 and 5 participants took the experiment on GLSL (Desktop), Unity3D (HMD and Desktop) and NVIDIA VRS (Desktop) platforms, respectively. The results of this experiment are shown in Figure 11. Our observation confirms the expected trend of increasing detection rates with the increasing multiplier. On the other hand, the detection rate for the predictions $\hat{\sigma}_s$ (multiplier = 1), is lower than 0.75 (1 JND), for which our predictor is calibrated. We identify several explanations for this. One possible reason is our custom loss function (Section 5) which penalizes overestimations with a larger weight in order to stay conservative. Another reason is the fact that the scene used in this experiment are much more complex, and the viewers are allowed to freely explore the scenes. Last, but not least, the tested setups include factors for which our model does not account. For example, in the HMD scenario, there is a significant additional quality degradation due to the lens distortions; in the VRS case, the rendering includes some non-negligible amount of temporal aliasing. Despite the above factors, it can seen that our model provides a rather consistent prediction of 75 % detection rate when a multiplier approx. 1-2 is used. This is a very satisfactory result, and we demonstrated in our further experiments that this prediction accuracy is sufficient for our applications.

7.3 Local vs. Global Adaptation

A suboptimal alternative to our method would be to adjust the foveal region radius in visual degrees, r, and the rate of resolution drop-off

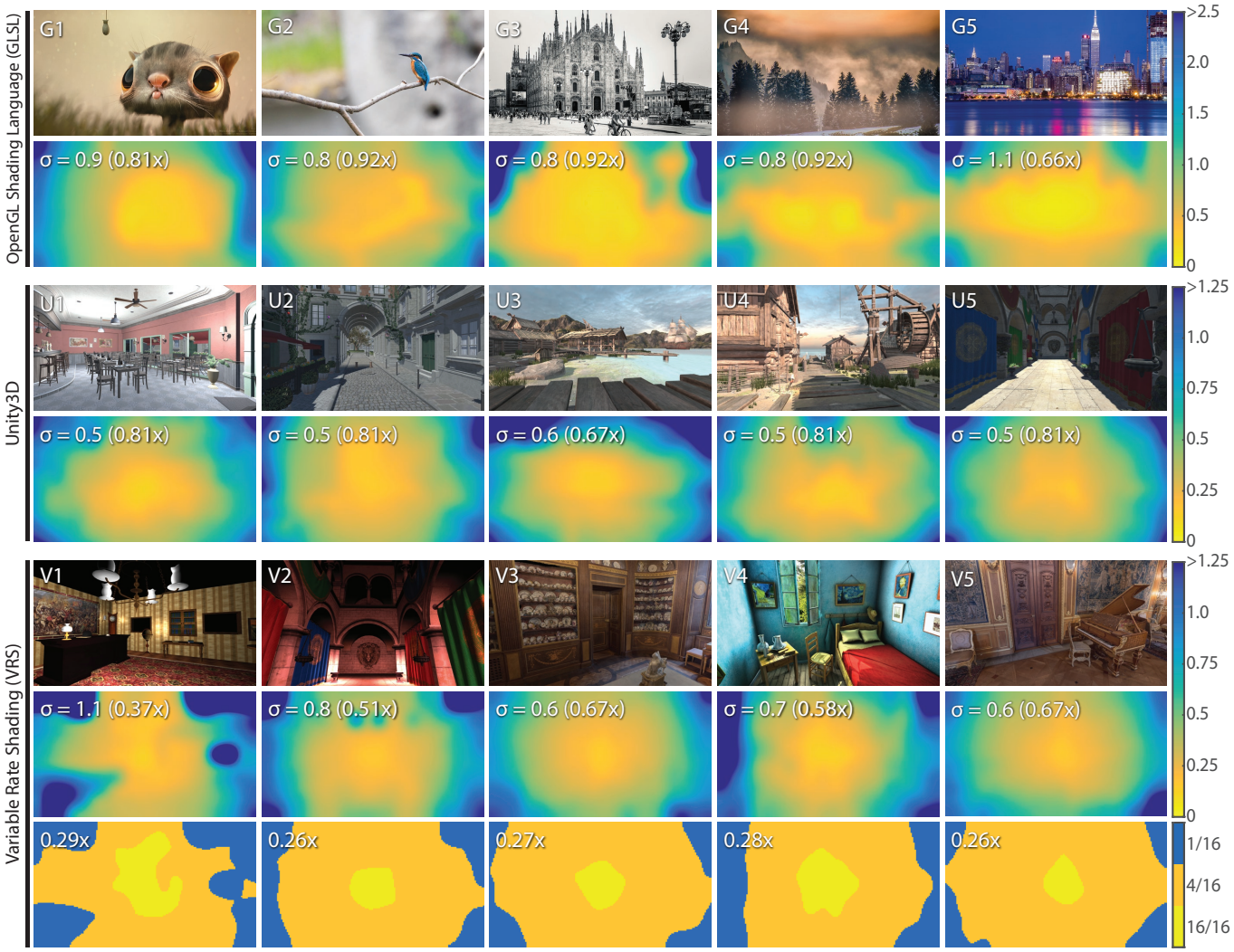

Fig. 10. Sample inputs and predictions from the implementation of our model on different platforms. The gaze position is assumed to be at the center of the screen for comparison. The color maps show the predicted standard deviations ($\hat{\sigma}_s$) in the second row for each platform. $\hat{\sigma}_s = 0$ represents a requirement for rendering in the native display resolution whereas larger values represent rendering in a lower resolution. Average value of each $\hat{\sigma}_s$ map is shown in the top-left corner and the numbers in parentheses are the estimates of the resolution reduction with respect to the standard foveated rendering. They are computed as the ratio of average sigma values from a standard foveated rendering implementation and our method. The standard foveation parameters are selected from the ground truth of text patch in our training set (Image 9 in Figure 5), which represents a visual content with high spatial frequencies. The third row in Variable Rate Shading shows the optimization of rendering resolution using our model, which is color coded according to different sampling rates supported by VRS API. The number in the top left corner represents the fraction of pixels which are actually shaded for each one of these frames.

Images G1-5 by Manu Jarvinen, Fxxu / Pixabay, the authors, analogicus / Pixabay and dawnfu / Pixabay, respectively, 3D models in U1 and U2 by Amazon, U3 and U4 by Unity, U5 and V2 by McGuire CGA [2017], V3 and V5 by The Hallwyl Museum / Sketchfab and V4 by ruslans3d / Sketchfab.

in the periphery, k, depending on the content of the whole frame (see Section 5 for definitions of r and k). This approach does not take into account local changes in the contrast; therefore, it is a globally adaptive foveated rendering. In a 2AFC experiment, we analyze the visual quality provided by our method (local adaptation) and the globally adaptive foveated rendering. At the beginning of the experiment, we run our method on input images which are provided in Figure 10 and compute the average standard deviation of foveation kernel $\bar{\sigma}_s$. Then we ask the participants to choose the optimal foveal region radius, $r \in \{4, 7, 11\}$, that provides the best visual quality in terms of overall sharpness for each image. At the end of this procedure, we compute the rate of resolution drop-off, k, as the value which gives the same average $\bar{\sigma}_s$ to have an equal amount of rendering cost in both methods.

Next, we asked the participants to compare local and global adaptation and choose the stimuli which offers the best overall sharpness.

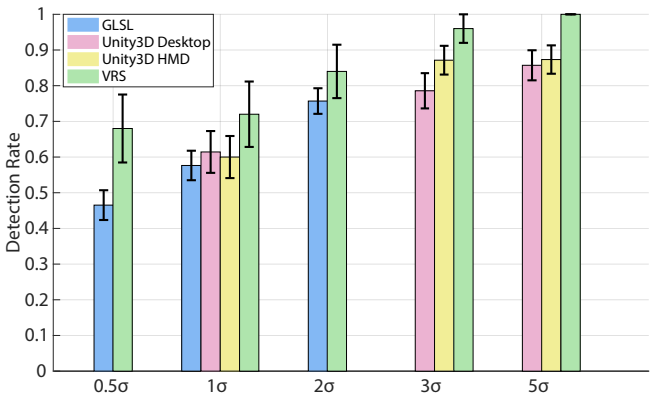

Fig. 11. Detection rates of the participants for our method and non-foveated rendering. x-axis represents different multipliers that we use for changing the average σ_s prediction to test the effect of different rendering budgets on the preferences of participants. The actual prediction of our method corresponds to the multiplier value of 1 and increasing values on the x-axis represent more limited rendering budgets. The trend in the detection rate shows that the participants actually detect the foveation and the detection rate for the actual rendering is smaller than 0.75 for all platforms when the predictions are not scaled. The error bars represent standard error.

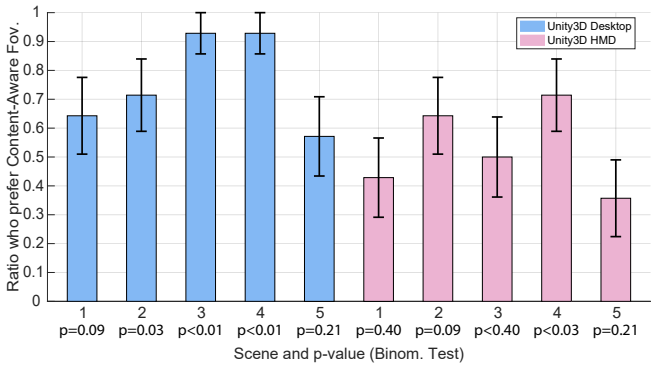

Fig. 12. The result of our subjective experiments where the participants compared our method with globally adaptive foveated rendering, which does not take local distribution of contrast into account. The error bars represent standard error.

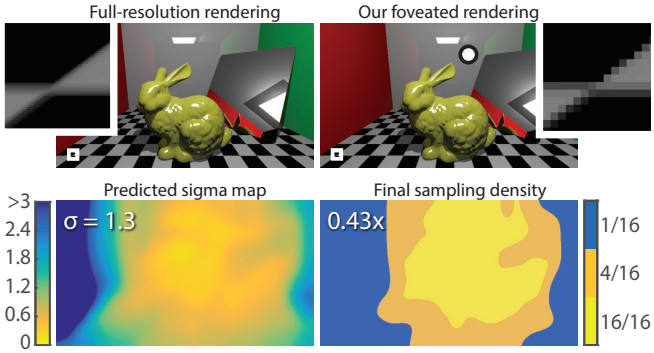

Fig. 13. On the top, the figure presents images generated by our ray tracer. On the bottom, we present the predicted map of standard deviations and the mask with sampling rates . The notation is the same as in Figure 10. Insets indicate magnification of the selected regions in the images depicted by the white rectangles. The white circle shows the gaze location.

This experiment was run on both HMD and desktop displays using our Unity3D implementation with gaze positions provided by the eye tracker. 14 participants took the experiment and a total of 70 comparisons were made on 5 scenes given in Figure 10 for each platform. As a result of this experiment, our method is preferred in 53 of 70 comparisons ($p < 0.01$, Binom. test) on desktop display and in 37 of 70 comparisons ($p = 0.28$, Binom. test) on HMD. Our analysis shows that the preference towards our method is statistically significant for desktop display while the preference for both methods are much closer on HMD.

The preferences of the participants for each scene are given in Figure 12. For all scenes tested on the desktop display, our method is preferred more often than global adaptation. On the other hand, the difference between two methods is significant for only one scene (U4, $p < 0.03$, Binom. test) on HMD. We think that the limited resolution of HMD and additional factors, especially those which affect the visual quality in the peripheral vision such as lens astigmatism, decrease the visibility of differences between two methods.

8 PROOF-OF-CONCEPT FOVEATED RAYTRACING

As a proof of concept, we implemented a custom foveated raytracer using OpenCL with RadeonRays routines, which was guided by our method. Our main goal was to test the overhead incurred by incoherent sampling and our predictor computation. In the first pass, a low-resolution image with 16-times fewer pixels is rendered. This image is then used by the predictor to estimate the standard deviations for the full-resolution image. In the second pass, we trace additional rays increasing the overall sampling such that it matches the prediction of our model. All the samples are then combined, and the final image is reconstructed using bilinear interpolation. An example rendering as well as predicted sigma and sampling rate maps are presented in Figure 13. As a result of applying our technique, 47 % of all rays was used to render the foveated version. For a comparison, a standard foveated rendering which would be below the detection threshold requires 55 % of all rays. Using our technique, the image from the figure was rendered in 15.87 ms on a PC with an NVIDIA RTX 2080 Ti graphics card in 2560×1440 pixels resolution. The rendering of the full resolution image took 22.70 ms, while the standard foveated rendering took 17.24 ms.

9 LIMITATIONS AND FUTURE WORK

In our work, we use Gaussian blur to model quality degradation due to foveated rendering. While this allowed us to derive a closed-form solution to the problem of finding the optimal foveation, it is only an approximation. We believe, however, that the accuracy of prediction will still hold for small deviations from this assumption, and when a better accuracy is needed, our method can be retrained in the future following the strategy proposed in the paper. Another exciting direction for future work is to consider temporal aspects

of foveated rendering. It is known that motion also reduces the sensitivity of the human visual system, and including this factor might be beneficial.

Our initial data collection procedure for calibration had a simpler design, where the stimuli consisted of a single patch displayed at a selected eccentricity from a pre-defined set in each trial. The process turned out to be prohibitively time-consuming, and it did not simulate well the case of foveated rendering because each patch was viewed in isolation on a uniform background. To improve the efficiency of data collection and to make the stimuli more realistic, we decided to perform experiments using stimuli filling the entire screen. This procedure allowed us to collect data simultaneously for an extensive range of eccentricity. Tiling the patches may introduce additional luminance frequencies which are not present in the original patch. We minimized this problem by flipping the patches. Furthermore, our experiment was performed for a limited set of (r, k)-pairs. Even though a denser sampling could lead to more accurate measurements, our procedure was sufficient to obtain a good model, which is confirmed in our validation.

With our current implementation, it is possible to reach a running time below 1 ms for computing the prediction from our model. Nevertheless, there is still room for improvement by using lower-level optimizations, especially for Laplacian pyramid decomposition, which is the most costly operation in the implementation. We believe that an implementation which is fully optimized for the hardware would achieve much shorter running times.

It is known that the total system latency, which is mainly determined by the display refresh rate and the eye tracker sampling rate, should be taken into account when testing novel foveated rendering techniques [Albert et al. 2017]. During our validation studies, our participants have not reported any artifacts (such as so-called "popping" effects or tunnel-vision) that could be directly attributed to the system latency. However, similar to other foveation techniques, we believe that our method would also require a less aggressive level of foveation in the presence of a noticeable amount of system latency unless a countermeasure is implemented [Arabadzhiyska et al. 2017; Griffith et al. 2018].

Vignetting and acuity reduction for off-axial viewing directions are typical problems in HMD optics. This enables further reduction in sampling rates when combined with standard and our content-dependent foveated rendering [Hoffman et al. 2018; Pohl et al. 2016]. For the optimal performance in our HMD system, such lens acuity falloff should be measured, which we relegate for future work.

During the experiments with VRS, we observed non-negligible temporal aliasing that this technology introduces. We do not account for such distortions in our model. In the future, it would be beneficial to extend our technique to consider such temporal artifacts to possibly avoid additional temporal aliasing which would be necessary to combat the problem.

Our model is currently targeting saving the computation by limiting the shading computation. We do not consider geometry pass which can have a considerable contribution to the overall rendering time. We believe, however, that our way of deriving the model, in particular, the experimental procedure and modeling, can be successfully used in the future to design more complete models for driving foveated rendering.

10 CONCLUSION

Recently proposed foveated rendering techniques use fixed, usually manually tuned parameters to define the rate of quality degradation for peripheral vision. As shown in this work, the optimal degradation that maximizes computational benefits but remains unnoticed depends on underlying content. Consequently, the fix foveation has to be conservative and in many cases, its performance benefits remain suboptimal. To address this problem, we presented a computational model for predicting a spatially-varying quality degradation for foveated rendering that remains unnoticed when compared to non-foveated rendering. The prediction is based on previous findings from human visual perception, but it is redesigned and optimized to meet the high-performance requirements of novel display technologies such as virtual reality headsets. In particular, a critical feature of our model is its capability of providing an accurate prediction based on a low-resolution approximation of a frame. Our validation experiments confirm that our technique is capable of predicting foveation which remains just-below the visibility level. This, in turn, enables optimal performance gains.

ACKNOWLEDGMENTS

We would like to express our gratitude to Michal Piovarči, Michał Chwesiuk and Itrat Rubab for their help. This project was supported by the Fraunhofer and Max Planck cooperation program within the German Pact for Research and Innovation (PFI), ERC Starting Grant (PERDY-804226), European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 642841 (DISTRO) and Polish National Science Centre (decision number DEC-2013/09/B/ST6/02270). The perceptual experiments conducted in this project are approved by the Ethical Review Board of Saarland University (No. 18-11-4).

REFERENCES

Hime Aguiar e Oliveira Junior, Lester Ingber, Antonio Petraglia, Mariane Rembold Petraglia, and Maria Augusta Soares Machado. 2012. *Adaptive Simulated Annealing*. Springer Berlin Heidelberg, Berlin, Heidelberg, 33–62.

Rachel Albert, Anjul Patney, David Luebke, and Joohwan Kim. 2017. Latency requirements for foveated rendering in virtual reality. *ACM Trans. on App. Perception (TAP)* 14, 4 (2017), 25.

Elena Arabadzhiyska, Okan Tarhan Tursun, Karol Myszkowski, Hans-Peter Seidel, and Piotr Didyk. 2017. Saccade Landing Position Prediction for Gaze-Contingent Rendering. *ACM Trans. Graph. (Proc. SIGGRAPH)* 36, 4 (2017).

Peter GJ Barten. 1989. The square root integral (SQRI): a new metric to describe the effect of various display parameters on perceived image quality. In *Human Vision, Visual Processing, and Digital Display*, Vol. 1077. Int. Soc. for Optics and Photonics, 73–83.

Peter GJ Barten. 1999. *Contrast sensitivity of the human eye and its effects on image quality*. Vol. 72. SPIE press.

M.R. Bolin and G.W. Meyer. 1998. A Perceptually Based Adaptive Sampling Algorithm. In *Proc. of SIGGRAPH*. 299–310.

Chris Bradley, Jared Abrams, and Wilson S. Geisler. 2014. Retina-V1 model of detectability across the visual field. *Journal of Vision* 14, 12 (2014), 22.

P. Burt and E. Adelson. 1983. The Laplacian Pyramid as a Compact Image Code. *IEEE Trans. on Communications* 31, 4 (Apr 1983), 532–540.

M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. 2014. Describing Textures in the Wild. In *Proc. IEEE Conf. on Comp. Vision and Pattern Recognition (CVPR)*.

Christine A. Curcio and Kimberly A. Allen. 1990. Topography of ganglion cells in human retina. *The Journal of Comparative Neurology* 300, 1 (1990), 5–25.

Andrew T. Duchowski, David Bate, Paris Stringfellow, Kaveri Thakur, Brian J. Melloy, and Anand K. Gramopadhye. 2009. On spatiochromatic visual sensitivity and peripheral color LOD management. *ACM Trans. on App. Perception* 6, 2 (2009).

Andrew T. Duchowski, Donald H. House, Jordan Gestring, Rui I. Wang, Krzysztof Krejtz, Izabela Krejtz, Radosław Mantiuk, and Bartosz Bazyluk. 2014. Reducing

visual discomfort of 3D stereoscopic sisplays with gaze-contingent depth-of-field. In *Proc. ACM Symp. on Appl. Perc. (SAP)*. 39–46.
Andrew T. Duchowski and Bruce Howard McCormick. 1995. Preattentive considerations for gaze-contingent image processing, Vol. 2411.
Joe Durbin. 2017. NVIDIA Estimates VR Is 20 Years Away From Resolutions That Match The Human Eye. https://uploadvr.com/nvidia-estimates-20-years-away-vr-eye-quality-resolution/. (May 2017). Accessed: 2019-01-10.
Henry Griffith, Subir Biswas, and Oleg Komogortsev. 2018. Towards Reduced Latency in Saccade Landing Position Prediction Using Velocity Profile Methods. In *Proc. Future Technologies Conf. (FTC) 2018*, Kohei Arai, Rahul Bhatia, and Supriya Kapoor (Eds.). Springer Int. Publishing, Cham, 79–91.
Brian Guenter, Mark Finch, Steven Drucker, Desney Tan, and John Snyder. 2012. Foveated 3D graphics. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 31, 6 (2012).
David Hoffman, Zoe Meraz, and Eric Turner. 2018. Limits of peripheral acuity and implications for VR system design. *Journal of the Soc. for Information Display* 26, 8 (2018), 483–495.
David Jacobs, Orazio Gallo, Emily Cooper, Kari Pulli, and Marc Levoy. 2015. Simulating the visual experience of very bright and very dark scenes. *ACM Trans. on Graph. (TOG)* 34, 3 (2015), 25.
Changwon Jang, Kiseung Bang, Seokil Moon, Jonghyun Kim, Seungjae Lee, and Byoungho Lee. 2017. Retinal 3D: Augmented Reality Near-eye Display via Pupil-tracked Light Field Projection on Retina. *ACM Trans. on Graph.* 36, 6, Article 190 (2017), 190:1–190:13 pages.
Petr Kellnhofer, Piotr Didyk, Karol Myszkowski, Mohamed M Hefeeda, Hans-Peter Seidel, and Wojciech Matusik. 2016. GazeStereo3D: Seamless disparity manipulations. *ACM Trans. Graph. (Proc. SIGGRAPH)* 35, 4 (2016).
Joohwan Kim, Qi Sun, Fu-Chung Huang, Li-Yi Wei, David Luebke, and Arie E. Kaufman. 2017. Perceptual Studies for Foveated Light Field Displays. *CoRR* abs/1708.06034 (2017). arXiv:1708.06034 http://arxiv.org/abs/1708.06034
G.E. Legge and J.M. Foley. 1980. Contrast masking in human vision. *Journal of the Opt. Soc. of America* 70, 12 (1980), 1458–1471.
J. Lubin. 1995. A visual discrimination model for imaging system design and development. In *Vision models for target detection and recognition*, Peli E. (Ed.). World Scientific, 245–283.
James Mannos and David Sakrison. 1974. The effects of a visual fidelity criterion of the encoding of images. *IEEE Trans. on Information Theory* 20, 4 (1974), 525–536.
Radosław Mantiuk, Bartosz Bazyluk, and Anna Tomaszewska. 2011a. Gaze-dependent depth-of-field effect rendering in virtual environments. In *Int Conf on Serious Games Dev & Appl.* 1–12.
Rafal Mantiuk, Kil Joong Kim, Allan G. Rempel, and Wolfgang Heidrich. 2011b. HDR-VDP-2: a calibrated visual metric for visibility and quality predictions in all luminance conditions. *ACM Trans. Graph. (Proc. SIGGRAPH)* (2011).
Belen Masia, Gordon Wetzstein, Piotr Didyk, and Diego Gutierrez. 2013. A survey on computational displays: Pushing the boundaries of optics, computation, and perception. *Computers & Graphics* 37, 8 (2013), 1012–1038.
Michael Mauderer, Simone Conte, Miguel A. Nacenta, and Dhanraj Vishwanath. 2014. Depth perception with gaze-contingent depth of field. In *Proc Human Fact in Comp Sys (CHI)*. 217–226.
Morgan McGuire. 2017. Computer Graphics Archive. (July 2017). https://casual-effects.com/data
Olivier Mercier, Yusufu Sulai, Kevin Mackenzie, Marina Zannoli, James Hillis, Derek Nowrouzezahrai, and Douglas Lanman. 2017. Fast Gaze-contingent Optimal Decompositions for Multifocal Displays. *ACM Trans. on Graph.* 36, 6, Article 237 (Nov. 2017), 237:1–237:15 pages.
Cornelis Noorlander, Jan J. Koenderink, Ron J. Den Olden, and B. Wigbold Edens. 1983. Sensitivity to spatiotemporal colour contrast in the peripheral visual field. *Vision Research* 23, 1 (1983).
Nvidia. 2018. VRWorks - Variable Rate Shading (VRS) website. https://developer.nvidia.com/vrworks/graphics/variablerateshading. (2018). Accessed: 2019-01-09.
Anjul Patney, Marco Salvi, Joohwan Kim, Anton Kaplanyan, Chris Wyman, Nir Benty, David Luebke, and Aaron Lefohn. 2016. Towards foveated rendering for gaze-tracked virtual reality. *ACM Trans. Graph.* 35, 6 (2016), 179.
E. Peli. 1990. Contrast in complex images. *Journal of the Opt. Soc. of America* 7, 10 (1990), 2033–2040.
Eli Peli, Jian Yang, and Robert B. Goldstein. 1991. Image invariance with changes in size: the role of peripheral contrast thresholds. *J. Opt. Soc. Am. A* 8, 11 (Nov 1991), 1762–1774.
Daniel Pohl, Xucong Zhang, and Andreas Bulling. 2016. Combining eye tracking with optimizations for lens astigmatism in modern wide-angle HMDs. In *Virtual Reality (VR), 2016 IEEE*. IEEE, 269–270.
Simon JD Prince and Brian J Rogers. 1998. Sensitivity to disparity corrugations in peripheral vision. *Vision Research* 38, 17 (1998).
Mahesh Ramasubramanian, Sumanta N. Pattanaik, and Donald P. Greenberg. 1999. A Perceptually Based Physical Error Metric for Realistic Image Synthesis. In *Proc. 26th Annual Conf. on Comp. Graphics and Interactive Techniques (SIGGRAPH)*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 73–82.
Martin Reddy. 2001. Perceptually optimized 3D graphics. *IEEE Comp. Graphics and Applications* 21, 5 (2001), 68–75.
L Ronchi and G. Molesini. 1975. Depth of Focus in Peripheral Vision. *Ophthalmic Res* 7, 3 (1975), 152–157.
Stephen Sebastian, Johannes Burge, and Wilson S. Geisler. 2015. Defocus blur discrimination in natural images with natural optics. *Journal of Vision* 15, 5 (2015), 16.
Mark Segal, Kurt Akeley, C Frazier, J Leech, and P Brown. 2013. The OpenGL Graphics System: A Specification (Version 4.4 (Core Profile) - October 18, 2013). (2013).
Michael Stengel, Steve Grogorick, Martin Eisemann, and Marcus Magnor. 2016a. Adaptive Image-Space Sampling for Gaze-Contingent Real-time Rendering. *Comp. Graphics Forum* 35, 4 (2016), 129–139.
Michael Stengel, Steve Grogorick, Martin Eisemann, and Marcus Magnor. 2016b. Adaptive image-space sampling for gaze-contingent real-time rendering. In *Comp Graph Forum*, Vol. 35. 129–139.
Hans Strasburger, Ingo Rentschler, and Martin Jüttner. 2011. Peripheral vision and pattern recognition: A review. *Journal of Vision* 11, 5 (2011).
Qi Sun, Fu-Chung Huang, Joohwan Kim, Li-Yi Wei, David Luebke, and Arie Kaufman. 2017. Perceptually-guided Foveation for Light Field Displays. *ACM Trans. Graph.* 36, 6, Article 192 (Nov. 2017), 192:1–192:13 pages.
Nicholas T. Swafford, José A. Iglesias-Guitian, Charalampos Koniaris, Bochang Moon, Darren Cosker, and Kenny Mitchell. 2016. User, Metric, and Computational Evaluation of Foveated Rendering Methods. In *Proc. ACM Symposium on App. Perception (SAP '16)*. 7–14.
Unity3D. 2018. Official website. https://unity3d.com/. (2018). Accessed: 2019-01-09.
Karthik Vaidyanathan, Marco Salvi, Robert Toth, Tim Foley, Tomas Akenine-Möller, Jim Nilsson, Jacob Munkberg, Jon Hasselgren, Masamichi Sugihara, Petrik Clarberg, et al. 2014. Coarse pixel shading. In *High Performance Graphics*.
Carlin Vieri, Grace Lee, Nikhil Balram, Sang Hoon Jung, Joon Young Yang, Soo Young Yoon, and In Byeong Kang. 2018. An 18 megapixel 4.3"1443 ppi 120 Hz OLED display for wide field of view high acuity head mounted displays. *Journal of the Soc. for Information Display* (2018).
B. Wang and K.J. Ciuffreda. 2005. Blur discrimination of the human eye in the near retinal periphery. *Optom Vis Sci.* 82, 1 (2005), 52–58.
Andrew B. Watson. 2014. A formula for human retinal ganglion cell receptive field density as a function of visual field location. *Journal of Vision* 14, 7 (2014), 15.
Andrew B. Watson and Albert J. Ahumada. 2011. Blur clarified: A review and synthesis of blur discrimination. *Journal of Vision* 11, 5 (2011), 10.
M. Weier, M. Stengel, T. Roth, P. Didyk, E. Eisemann, M. Eisemann, S. Grogorick, A. Hinkenjann, E. Kruijff, M. Magnor, K. Myszkowski, and P. Slusallek. 2017. Perception-driven Accelerated Rendering. *Comp. Graphics Forum* 36, 2 (2017), 611–643.
Wenjun Zeng, S. Daly, and Shawmin Lei. 2000. Point-wise extended visual masking for JPEG-2000 image compression. In *Proc. Int. Conf. on Image Processing*, Vol. 1. 657–660.
Wenjun Zeng, Scott Daly, and Shawmin Lei. 2001. An Overview of the Visual Optimization Tools in JPEG 2000. *Signal Processing: Image communication Journal* 17, 1 (2001), 85–104.

A STIMULI CHOICE

We diversify our dataset of stimuli by choosing 18 patches from a large collection of 5640 images. We maximize the dissimilarity between them by choosing every patch in a greedy fashion using the following formula:

$$I_{n+1} = \arg\max_{I} \{ d(I, \mathcal{D}_n) \}, \tag{21}$$

where $\mathcal{D}_n$ is the dataset consisting of the first n patches and I_{n+1} is the next patch that is added to the dataset. The dissimilarity measure $d(I, \mathcal{D}_n)$ between a candidate image and the dataset is defined as

$$d(I, \mathcal{D}_n) = \sum_{k=1}^{K} \left| \mathcal{L}_k(I) - \frac{1}{|\mathcal{D}_n|} \sum_{I_d \in \mathcal{D}_n} \mathcal{L}_k(I_d) \right| \tag{22}$$

where $\mathcal{L}_k(I)$ is the mean absolute deviation of pixels in kth level of Laplacian pyramid for image I. This dissimilarity metric maximizes the diversity of frequency content in the dataset by picking the image which has the Laplacian decomposition least similar to the average of existing images.