ZACHODNIOPOMORSKI UNIWERSYTET TECHNOLOGICZNY W SZCZECINIE

WYDZIAŁ INFORMATYKI

mgr inż. Marek Wernikowski

Generation of images for stereoscopic displays using selected perceptual features of human visual system

Dissertation

Generowanie obrazów dla wyświetlaczy stereoskopowych z uwzględnieniem wybranych cech percepcyjnych układu wzrokowego człowieka

Rozprawa doktorska

Promotor: dr hab. inż. Radosław Mantiuk, prof. ZUT

Szczecin 2022

Contents

Introduction

Recent years have brought a significant increase in the interest of equipment manufacturers in imaging technology. The most glaring example of this phenomenon is the vast increase in demand for *virtual reality goggles* (VR goggles). They are an example of near-eye displays – by displaying separate images for both eyes, they create a *stereoscopic effect* activating vergence depth cue. The new advancements in this field provide higher and higher resolutions, increasing the realism and immersion of the rendered scenes. Other examples of near-eye displays are *multi-focal displays*. This technology displays the images on several planes for each eye, making it possible to activate the accommodation cue.

Problem definition

Both of the mentioned technologies, despite the visualisation improvements, impose additional performance and quality requirements on image rendering methods. Thanks to the constant increases in the efficiency of processors and graphics cards, it has become possible to create computer games with previously unattainable graphics quality shown in real-time. Currently, the most popular rendering method is scan-line rasterisation. A rapid growth in hardware performance allowed developers to use other technologies which were previously too slow to be used in real-time, such as ray tracing. This technology can create photo-realistic rendering using the accurate modelling of light and its interactions with different surfaces. However, current scan-line and ray tracing implementations rely on approximation methods to limit the spatial and temporal sampling frequencies, even for standard displays.

Due to their specificity, VR displays need higher temporal frequencies than typical displays. This requirement entails performing the same operations faster than what was previously sufficient. Stereoscopic displays require a simultaneous generation of two images (one per eye), while for displays with accommodation it is necessary to generate multi-view light fields. The near-eye displays have a much lower pixel density than typical displays, as the screen is right in front of the eyes. These displays are also prone to lens distortions, which increase the size of some

pixels even more. It causes a more visible aliasing problem. It is exacerbated by the virtual camera's constant changes in its position due to head movements. The resulting temporal aliasing significantly reduces the quality of the displayed image. Therefore, an anti-aliasing method is needed, which causes further computational costs.

Proposed solutions

The solution to leverage these issues is to *utilise the features of human vision in rendering*. Including perception in visualisation would reduce the complexity of computations and reach higher performance levels while keeping the image quality close to the ground truth. In this work, the ways of including the selected perception mechanisms in the image generation are proposed. Depending on their nature, they can be used either to improve the rendering quality or to increase the performance without introducing visible changes to the images.

The main analysed mechanism is *vision acuity*. Human vision acuity is the highest at the circular area around the gaze point with a radius of 1-2 visual degrees [60]. Outside this region the quality of vision drops rapidly, reducing the visibility of various scene features. Such a phenomenon causes a decrease in visible spatial frequencies in the peripheral vision region. Because of that, reduction in the number of such frequencies to a certain point would not result in any visible quality differences. Another effect caused by reduced acuity is the contrast sensitivity: the further an object is from the gaze point, the harder it is to differentiate between two shades of colour. The last effect is visual crowding – when the stimulus is presented close to other stimuli, it is harder to pick it out than when it is presented on its own. This effect can only be observed from the peripheral region of vision. A certain level of distortions can be applied to such stimulus without causing any visible differences. By analysing an image in regard to the effects caused by differences in visual acuity it is possible to determine the reduction in quality that would not be invisible to the human observer. In this work, it is determined, what is the acceptable and imperceptible level of distortions depending on the rendered content and available hardware.

Thesis and goals of the dissertation

The thesis of the dissertation is formulated as follows:

Considering the perceptual features of the human visual system improves the efficiency of the synthesis of stereoscopic images while maintaining a comparable perceptual quality of these images.

The goals achieved to prove this thesis are:

- The development of a method that uses the gaze direction and the sensitivity to contrast features in the human visual system to accelerate the synthesis of images (see Chapter 2 and 3).
- The development of a method for accelerating the generation of images to be displayed on stereoscopic displays with accommodation while maintaining their perceptual correctness (see Chapter 4).
- The development of a method of perceptual image filtering in the time domain to reduce aliasing on stereoscopic displays (see Section 2.4).

Research methodology

The research was carried out using experimental and analytical methodology. Selected components of the human visual system were analysed to determine their limitations and possible use in rendering and visualisation systems. This analysis was often accompanied by perceptual experiments measuring the features of the human visual system. The developed methods of synthesising and displaying images have been modelled, implemented and tested. The tests were performed based on perceptual experiments in the form of user studies.

Dissertation content

Decrease of spatial frequencies visibility and contrast sensitivity in the peripheral vision region softens the constraints on the image quality. In Chapter 2 the foveated rendering technique is introduced, in which mentioned features of the human visual system are used. By tracking the gaze location of the observer, it is possible to determine the angular distances to all image regions, allowing the calculation of maximal acceptable resolution reduction. This approach results in performance improvements without affecting the perceived image quality.

Decreased visual acuity in peripheral vision region does not result solely in the resolution reduction – visual crowding results in uncertainties of peripheral object structures. Additionally, the studies have shown [70] that the lack of high frequencies caused by resolution reduction may be noticeable. As such, the problem of perceptual rendering is extended to the idea of *metamers* - images with different pixel content that look identical to human eyes. Metamers are usually defined for specific eccentricities. It is because of a simple observation: while looking at the centre of an image in high definition, slight modifications to the points on the edges

would stay invisible to the human observer. So the question arises: to what extent is it possible to remove irrelevant picture details while keeping it indistinguishable from the original? Research in this area is described in Chapter 3.

High-frequency image sampling is needed only to the extent to which the human visual system is sensitive. This statement is particularly important for multi-focal display image reconstruction, as its rendering is considerably more expensive than for standard display. The process of decomposition – dividing rendered images between image planes – is the most expensive part because it typically requires rendering from multiple points of view. The simplest way to approximate this process is to decompose the image using only the depth of objects in the scene. Unfortunately, such a simple method fails at the edges of objects and occlusion areas. A new hybrid method is proposed, which combines mentioned techniques using human perception. It detects which places require more complex approaches and which do not. Thanks to this, one can significantly improve the performance of image decomposition calculations without decreasing quality. A description of the method leading to such an effect can be found in Chapter 4.

Another mechanism described in this work is the temporal adaptation to different lighting conditions. Since this process is not instantaneous, a model is proposed based on the human visual system. Its architecture is presented in Section 2.3, where it is also compared to simpler approaches.

In the following Chapter (Chapter 1) concepts related to human perceptions are introduced. The methods described in the remaining chapters are based on these basic mechanisms of the human visual system.

Chapter 1

Background

This chapter provides the definitions of basic terms used throughout the dissertation. In particular:

- Visual acuity (Section 1.1) – included in the nonuniform image sampling, which is required for *foveated rendering* (see Chapter 2),
- Luminance and brightness (Section 1.2) – used to determine the visual adaptation level to model *adaptation to brightness* (see Section 2.3),
- Contrast (Section 1.2) – calculated to measure the acceptable changes to the image without affecting its perceptible quality to make the foveated rendering process *contrast aware* (see Chapter 2),
- Depth cues (Section 1.3) – used to create *depth perception* for multi-focal displays, which are explained in Chapter 4,
- Metamers (Section 1.5) – the concept is used to determine what types of distortions are invisible to the human visual system (see Chapter 3).

1.1 Visual acuity

The *visual acuity* of the human visual system is typically measured by evaluating the highest detectable *spatial frequencies* – frequencies of sinusoidal patterns oscillating between darker and brighter colours. The minimum frequency for which both minimum and maximum values of such a pattern are resolved by the visual system as one colour is called the *detection threshold*. The capabilities of the visual system are defined by the *photoreceptors*.

Photoreceptors collect the light rays on the eye retina, where they are converted into an electrical signal by opsin and rhodopsin: light-sensitive proteins [66].

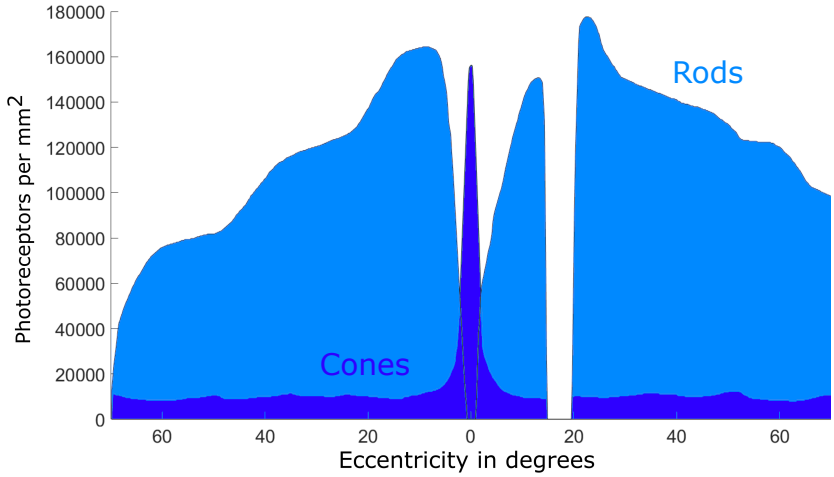


Figure 1.1: The distribution of cones and rods in the human retina [56].

The signal is then transferred to the visual cortex, where the final experienced image is created.

The critical stage is the gathering of photons by the photoreceptors. Their structure and placement offer several advantages and some limitations. In this chapter, examples of such features are shown.

There are two main types of photoreceptors: *cones* and *rods*. A graphic showing their approximate distribution on the retina is shown in Figure 1.1.

Cones are responsible for detecting colours. They reside in the fovea where their count is 125500 $cells/mm^2$ [31]. In the peripheral region, their number drops significantly, down to 2000-3000 $cells/mm^2$. Because of their large concentration at the centre of vision, they provide high acuity. They are also swift to respond to light changes. However, they have lower sensitivity and require more light than rods to remain active. They provide *photopic vision* which happens when the eye is in well-lit conditions. Cones can be separated into three types, responding to three different wavelength ranges: S, M, and L, centred in colours blue, green, and red correspondingly.

Rods are the second type of photoreceptors, responsible for *scotopic vision* – seeing in the dark. As opposed to the cones, they reside outside the fovea. At the distance of 4 mm from the fovea, they reach the amount of 90000 $cells/mm^2$ [31]. Their number slowly decreases towards the periphery, where it flattens at 30000-40000 $cells/mm^2$. The total number of rods is much higher than cones – in each retina, there are about 120 million rods and 6 million cones. That makes them very sensitive, enabling the capture of even single photons to create vision. Rods come in one type only. Because of that, they cannot create a colour vision and only produce a monochromatic image. The distribution of cones influences the resolution

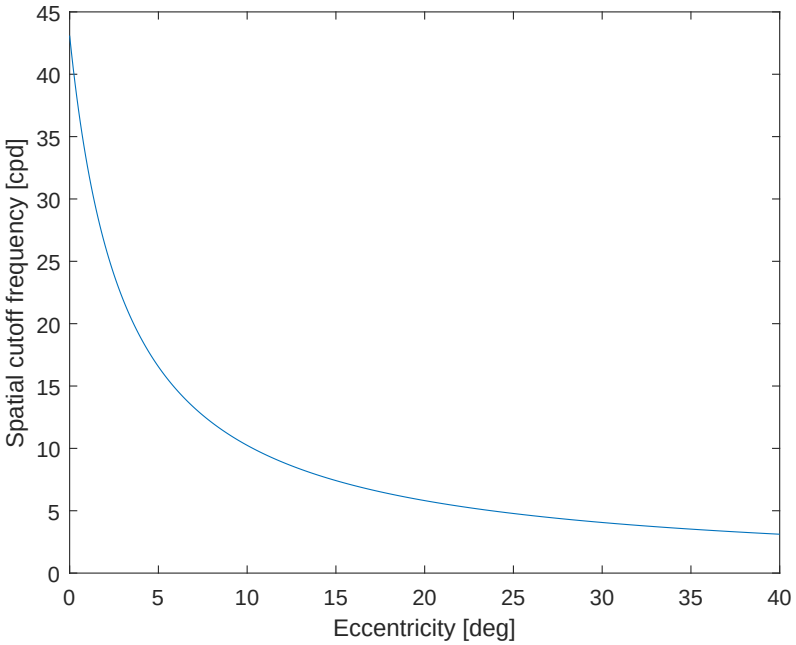


Figure 1.2: The cutoff frequency function of human visual system (reused image from own publication [80]).

of vision. It can be measured using *spatial cutoff frequency* – a function that shows the highest spatial frequency that can be resolved by human visual system at a given eccentricity. Loschky et al. [49] define it as:

$$f_c(d) = 43.1 \cdot \frac{E}{E + d}, \tag{1.1}$$

where d is the eccentricity in visual degrees, and E is the eccentricity at which the spatial cutoff frequency drops to half. It can be approximated to $E = 3.118°$. The plot of this function is presented in Figure 1.2.

An area in the distance between 0° and 1.5° eccentricity is called *a foveal region*. It provides the highest level of visual acuity [14]. The farther area, in which the quality of vision is lower because of the radically decreased number of cones, is called *a peripheral region*.

1.2 Contrast and brightness

Luminance is defined as an amount of light passing through a certain point in space. In the display context, it is calculated for every pixel individually as the amount of light it emits. *Brightness* indicates the subjective luminance perceived by an observer.

Contrast, depending on the context, can be defined in various ways. In

general, it is defined as a difference between the brightness of two objects. In this work, a definition of *Weber contrast* is used. It is given by the following formula:

$$C_w = \frac{L - L_b}{L_b}, \tag{1.2}$$

where L is the luminance of the analysed feature, and L_b is the average luminance of the background surrounding it. The ability to detect contrast depends on the contrast sensitivity function [60]. It measures the maximum detectable contrast based on the spatial frequency of the observed signal.

The visual system can adjust to various lighting conditions thanks to visual adaptation. Human visual system is able to create a detailed image at wide range of luminances, including night sky ($10^{-6}\frac{cd}{m^2}$) and sun-lit environments ($10^{8}\frac{cd}{m^2}$) [57, 61]. The adaption to light happens in the retina due to the functioning of photoreceptors located there [32]. The luminance that the eye is adapting to at a given moment is called *adaptation luminance*. The luminance of all objects in the field of view provides a base for its value, though the objects closer to the foveal part of vision have a higher impact. As the direction in which the eyes are turned changes constantly, the retina is in *maladaptation* state – it is trying to reach the adaptation luminance level. However, because of its rapid changes, it never does.

1.3 Depth cues

Depth perception is an essential trait of the human visual system, which estimates the distance to visible objects. The means of assessing this information are called *depth cues*, which are helpful in everyday life, e.g. in avoiding obstacles and grabbing objects. Two such cues directly caused by the physical properties of human eyes are accommodation and vergence.

Accommodation is the process of changing the physical shape of lenses in the eyes to change their focal point. The light rays coming from a specific point cross at one point on the retina after passing through the lens. Whenever one changes their gaze location, the lens changes its shape accordingly, keeping the observed object in focus. Human capabilities of accommodation decrease with age, as the minimal focus distance drops from 6.5 centimetres for children to over 1 meter for elders [48].

Vergence is the action of rotating the eye in the direction of the observed object. As each eye has a slightly different view of the observed scene, this action is required to connect both images into one. Vergence prevents the double vision of the focused object. It could still occur whenever one focuses on a different distance than the object is located.

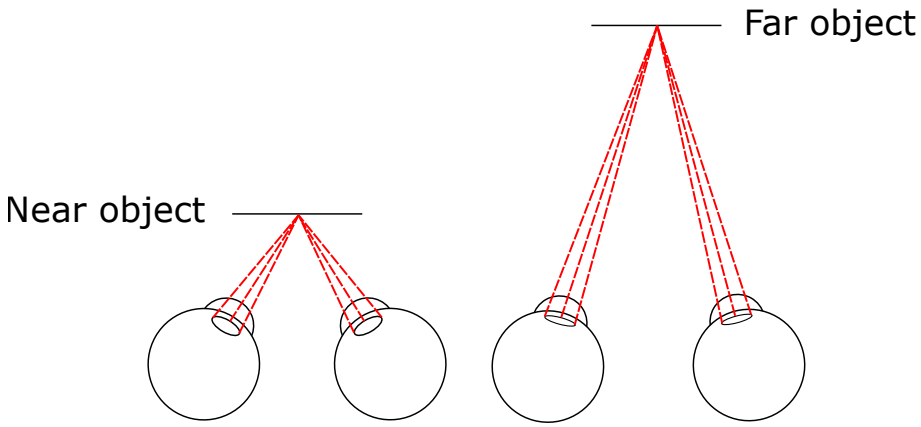
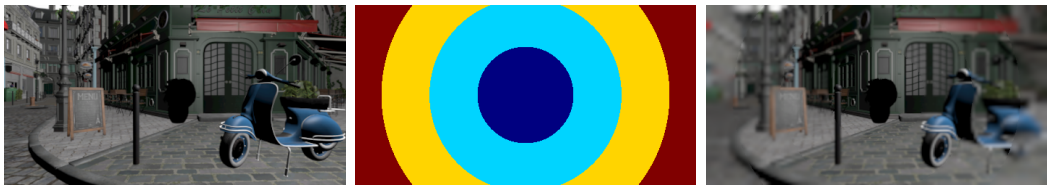


Figure 1.3: Eyes focused on the near and far objects. Notice the difference in rotations and lens shapes of the eyes.

The combination of accommodation and vergence plays an important role in depth perception, as demonstrated in Figure 1.3. Their combined usage is needed for correctly assessing the distances between the objects as well as the observer's position. Chapter 4 shows a way of using both cues in the rendering process.

1.4 Foveated rendering



(a) Full resolution image (b) Foveation mask (c) Foveated image

Figure 1.4: An example of standard foveation procedure. Image (a) shows the original, unmodified frame. Image (b) contains the foveation mask that defines sampling rates for different image areas. Every sampling rate is marked with a different colour. Image (c) presents the final image, where every region was generated with a sampling rate corresponding to the mask.

Foveated rendering is a replacement for standard rendering to reduce the computation cost. Perceptually irrelevant regions of the final image can be rendered in lower resolution without any visible quality degradation. The basis of such a rendering system is a *resolution map* containing the information about the required acuity in each image fragment (see Figure 1.4). The image synthesis can be parameterised by knowing the map's contents before rendering. Therefore, the resolution and level of detail could be based on the values on the map. If the cost of calculating the map is negligible, such a process should lead to performance improvements

in every case. The assumption of fast and low-cost map calculation is crucial for designing the whole rendering system.

Standard foveated rendering generates a map in the static form. It uses the universal features of the human visual system to evaluate the general best formula of undetectable spatial frequency reduction. The acuity of vision deteriorates with the increase of eccentricity [60], therefore the values representing required resolution are the highest at its centre and decrease in the peripheral region. By tracking the eye movement using an eye-tracker, the map may follow the gaze to keep its centre exactly at the gaze position. The most important aspect of this technique is the parameters of resolution computation as a function of eccentricity. Such studies have already been performed ([23], [58], and [67]) and they lead to the reduction of shading operation count by up to 70%.

1.5 Metamers

Every type of cone interacts with a specific range of wavelengths. A response of a cone is influenced by the sum of all light it receives. Because of that, the same response can be created for differently distributed light waves with the same total intensity. Such a phenomenon is called metamerism. *Metamers* are defined as colours created by such waves that will look identical to the human visual system. Recently this term has been extended to include collections of perceptually indistinguishable colours and images.

The visual receptive field (area supplying the information for a single neuron) increases with eccentricity [17]. As shown by Freeman et al. [19], large receptive fields force the photoreceptor to convert a large number of photons to a nerve impulse through spatial pooling. Such a phenomenon leads to a loss of information that cannot be reversed. From this observation, it can be deduced that multiple variations of photons stream create the same response in the brain. In other words, for each image generated in the brain based on the observed content, other images look the same for the human visual systems. Such images are metamers.

The density of photoreceptors decreases with eccentricity. They reach their maximum in fovea centralis, which is the central region with a 1.5 mm radius at 0° eccentricity [73]. Here perceived light creates the highest visual acuity with fine details and small receptive fields. As such, there is almost no room for creating new metameric images. On the other hand, larger receptive fields in peripheral regions increase the possibility of distorting the images without disturbing the observer.

1.5.1 Vision distortions

The resolution of vision is not uniform along the whole field of view. While objects looked at directly appear sharp, the rest of the view is usually distorted. There are two main reasons for such phenomenon: the lens behaviour and photoreceptors density.

Unlike standard camera lenses, lenses in the eyes can adjust their shape, changing their focal point. That enables focusing on objects at varying distances. However, changing the focal point as a consequence blurs the objects closer and further from the object in focus. The blur magnitude of the observed object depends on its distance from the object in focus.

The lens's optical properties can solely explain this change, and, as such, it does not introduce any other types of distortions. The blurring may be used for performance improvements in the multi-focal display rendering. Standard displays and virtual reality headsets cannot benefit from it, however. For standard display, the focal point position depends on the distance to the screen. In a virtual reality headset, the lenses are constructed with a specific focal point so that the user is always focused on one point, typically outside of the headset [36].

Blur caused by lenses differs in effect from the photoreceptors' distortions. The blur originates from lens optics. The distortions, however, are caused by the layout of the photoreceptors. The imaginary uniform layout of photoreceptors would lead to sharp vision over the entire focused visual field, similar to the camera's output. The actual non-uniform formation creates gaps in the vision, which have to be filled by the brain. The way the samples are connected remains undiscovered – it is unknown how the brain processes the samples nor how it behaves with various observed objects. However, it is known that the brain can create an image from a low number of samples that do not contain any visible discontinuities in the peripheral region.

Chapter 2

Perception-driven Rendering

This Chapter describes the process of adjusting the rendering pipeline to account for specific features of the human visual system. The author's contributions in various project stages are listed at the end of the Chapter (see Section 2.5).

2.1 Objectives

The current rendering systems can be extended by including the perception in two ways:

- improving the performance by rendering only perceptible content and excluding costly operations which do not have any visible effect,
- improving the quality by including the visual effects introduced by photoreceptors.

Even though current foveation solutions create significant performance improvements (see Section 1.4), they could be further improved. The scene content is the most crucial aspect not considered in the existing studies. Because of visual masking, some image elements can be hidden from the visual system, even if they are close to the gaze position. Furthermore, the contrast sensitivity is not a constant value but a function changing with eccentricity. Therefore, a new foveation procedure is needed that would include the perceptual features in the image synthesis.

In Section 2.2, the idea of the global foveation model is extended by introducing local features of the rendered scene. Thanks to that, the performance can be improved by varying the resolution reduction based on the content.

One of the most apparent visual phenomena that is an essential part of perception is visual adaptation. The process of adjusting the sensitivity of rods and cones allows the perception of scene details in almost any lighting condition. However, adjusting to different luminance levels can take a long time in certain situations.

An example of long adaptation is coming into the darkroom from the sun-lit outside environment. In Section 2.3 the idea of simulating the process of adaptation is explained. Additionally, a study is presented in which its usability was evaluated in computer games and compared to its alternatives.

Even though the foveated rendering dramatically improves the performance, it has its downsides. One of the biggest problems significantly increased by lowering the resolution in the peripheral regions is temporal stability. Even though human eyes might not be able to see the details in the periphery fully, temporal distortions and rendering artefacts can still be perceptible [70]. Because of that, additional measures have to be taken to solve this problem. Unfortunately, standard temporal anti-aliasing methods are costly operations. In Section 2.4 a method of detecting aliasing is proposed, which would allow executing anti-aliasing on predetermined pixels only.

2.2 Foveated rendering system

2.2.1 Content-aware foveated rendering

This chapter describes the details of creating a custom resolution map to perform foveated rendering. It also explains the experimental procedures performed to confirm the model's viability. More information on it is present in the publication in which this project has been a subject of: [71].

Existing foveation solutions typically use statically generated resolution maps (see Section 1.4). As a way of including perception in the rendering process, a new method is proposed. As opposed to the standard method, the resolution map is generated by the *predictor*, modelled to calculate the required resolution in every frame separately. The values generated by the predictor are based on the limitations of human visual system: visual acuity (Section 1.1) and contrast (Section 1.2). For that, precise extrinsic data about the viewing conditions are needed, such as observation parameters (distance to the screen, display dimensions, pixel size, eccentricity) and the presented scene content. After measuring such parameters, the acceptable resolution reductions are estimated. A visualisation of such a system is presented in Figure 2.1.

The proposed approach to fetch the content information is the rendering of a *low-resolution frame* before other parts of the rendering process. The dimensions of such a frame are reduced four times compared to the original. That way, the general scene content would be known beforehand and could approximate the quality requirements in each image region. Nonetheless, this method has limitations because

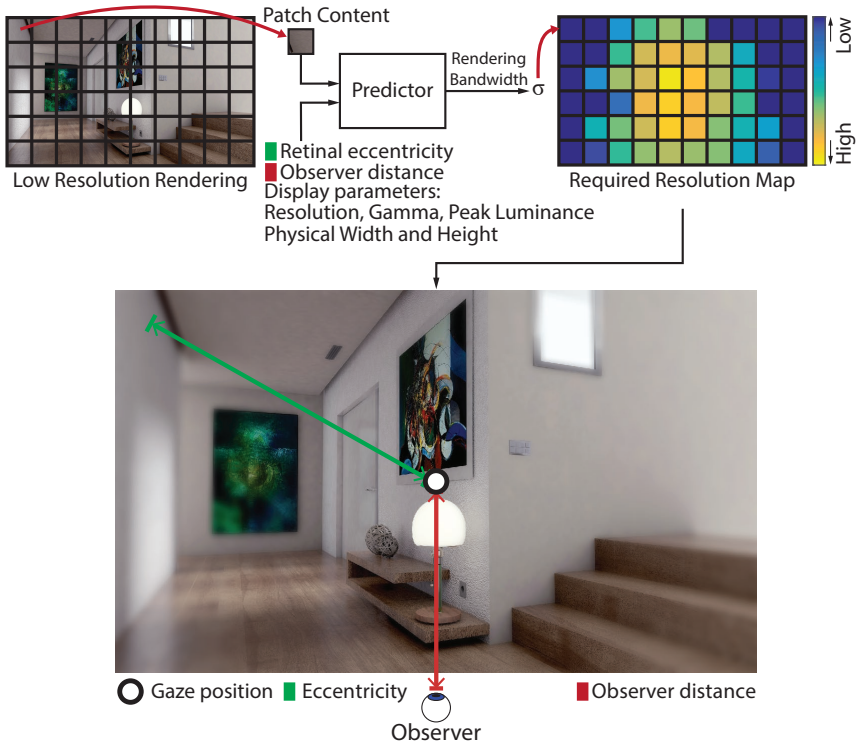


Figure 2.1: The overview of the method (reused image from own publication [71]).

the amount of gathered content is incomplete compared to the high-resolution rendering. To correctly measure how much information is lost and how much of it is needed, an experimental procedure is run, explained in more detail in Section 2.2.4. The method does come with the extra cost needed to render the reduced image. However due to a low number of samples it is only a fraction of the whole frame generation time.

The initial low-resolution frame is divided into 32×32 image patches for whitch the predictor calculates the resolution map separately. The whole inference process is divided into two stages. The goal of the first stage is to convert the values of each image patch into *perceptual contrast*. The possibilities of noticing contrast can change based on multiple factors. The predictor's goal is to consider them all and convert raw image contrast values into the contrast visible by a human.

The second part of the predictor involves changing the perceptual contrast into the measure of indistinguishable resolution reduction. It is used to provide the minimum resolution for each fragment of an image. Then such values may be directly employed in the rendering algorithm to generate images using fewer computational resources than full-resolution rendering.

2.2.2 Perceptual contrast

Calculation of perceptual contrast takes a few steps shown in Figure 2.2. Each block represents a single step that encodes perceptual data into the output values. In this Section, more detail is given explaining their purposes.

Intensity to Luminance Conversion To work directly on photometric units, the image must be converted into luminance space. The display parameters were measured using a luminance meter to calculate the gamma curve. The displayed RGB image was transformed into the luminance values using those parameters.

Band Decomposition The next step is the calculation of luminance contrast. Since the calculations are based on human perception, the contrast measurement has to be directly related to how the eyes perceive luminance. To this end, the definition of local band-limit contrast was used, proposed by Peli [59]. Following this definition, an image is divided into spatial frequency bands, in which the contrast values are calculated separately. This way of calculating contrast is motivated by the fact that contrast sensitivity is different for various spatial frequencies [12]. It may be determined locally for each pixel using the following equation:

$$c(x,y) = \frac{a(x,y)}{l(x,y)}. \tag{2.1}$$

Arguments x and y define the pixel position. a is the analysed band-limited image in the frequency domain, multiplied by the inverse Fourier transform, which is also band-limited. l is the filtered image containing only the information about spatial frequencies with lower values than in the band limits ($l(x,y) > 0$). In this work, the following general formula was used, which was applied to each analysed frequency band:

$$C(f,p) = \frac{\Delta L(f,p)}{L_g(f,p) + \epsilon}. \tag{2.2}$$

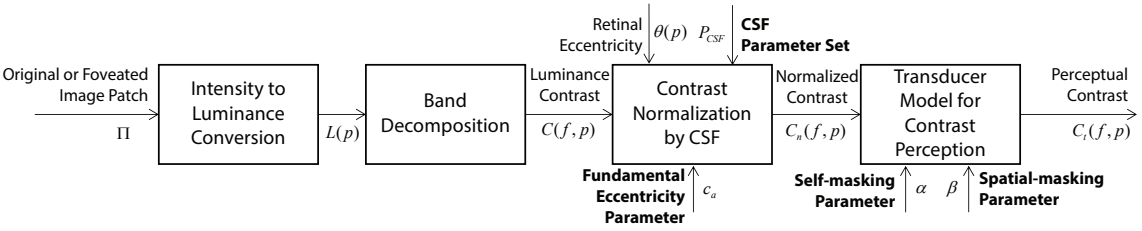


Figure 2.2: The procedure required to calculate perceptual contrast for each image patch. The parameters shown in bold are constant values gathered from the perceptual experiment explained in Section 2.2.4 (reused image from own publication [71]).

Parameter f determines the spatial frequency of the analysed pixel p. An image was divided into spatial frequency bands using the Laplacian pyramid [11]. Each pyramid layer contains spatial frequencies spanning one octave. p defines the pixel position (i.e. the x and y coordinates from the 2.1 formula). $\Delta L(f,p)$ is a single value in the Laplacian pyramid, corresponding to the difference in luminance between two layers. It is equivalent to the $a(x,y)$ parameter in the 2.1 equation. L_g (corresponding to l in 2.1) is calculated based on the Gaussian pyramid from the original image. The pyramid level two levels below the current Laplacian pyramid level was used. Such images are increased four times in each direction using linear interpolation to extend their size to the size of an image from the analysed layer. Consequently, that image contains all frequency values two octaves below the original image. Because a possible scenario exists in which $L_g(f,p) = 0$, a small constant value ϵ was added, protecting against division by zero problems.

Contrast Normalisation by CSF Obtained luminance contrast value in each pixel does not consider additional aspects of human vision. It is accurate for looking at an image directly (i.e., it is visible in the highest acuity) and ignoring the visual adaptation. It is, however, an oversimplification of the natural phenomenon. Every image could span a large display area, so eccentricity information is needed to calculate perceptual contrast. Additionally, because eyes adapt rapidly to changing lighting conditions, the contrast sensitivity includes the adaptation luminance, i.e. the mean luminance of the region in proximity to the analysed pixel. With the mentioned aspects of vision, the actual level of perceived contrast was calculated by normalising the luminance contrast using the contrast sensitivity function (CSF):

$$C_n(f,p) = C(f,p) \cdot S_{CSF}(f, \theta(p), L_a(f,p)). \tag{2.3}$$

S_{CSF} is an eccentricity-dependent CSF that weights the contrast to make it dependant on human perception. It receives three parameters: spatial frequency f, eccentricity θ calculated as a distance in visual degrees from the gaze point, and $L_a(f,p)$, which is adaptation luminance.

Eccentricity-dependent sensitivity function extends contrast sensitivity function by the eccentricity information. It is calculated based on the contrast threshold (minimal contrast visible in the given conditions) defined by Peli et al. [60] in the following way:

$$T(\theta, f) = A \ \exp(c_a \theta f). \tag{2.4}$$

c_a is a constant value called *fundamental eccentricity parameter*. A is a contrast

threshold measured during the direct observation of the object - in other words, it is a contrast threshold value which does not take into account the eccentricity ($T(0, f) = A\ \exp(0) = A$).

The value to compute is contrast sensitivity defined as an inverse of contrast threshold. Therefore the formula can be presented as follows:

$$\frac{1}{T(\theta, f)} = \frac{1}{A} \cdot \frac{1}{\exp\left(c_a \theta f\right)}. \tag{2.5}$$

In that case, $\frac{1}{A}$ is standard, non-eccentricity-dependent CSF, and $\frac{1}{T(\theta,f)}$ is eccentricity-dependent contrast sensitivity (which from now will be noted as S_{CSF}).

In order to extend the CSF function by including the adaptation luminance, the right-hand side of the equation 2.5 is multiplied by $a(L_a)$, which is the result of the function defining the relationship between adaptation luminance L_a and peak contrast sensitivity at the analysed image pixel. For that purpose, the equation proposed by Barten [6] is used:

$$a(L_a) = (1 + 0.7/L_a)^{-0.2}. \tag{2.6}$$

To sum up, a complete formula for calculating the normalised contrast is as follows:

$$S_{CSF}(f, \theta, L_a) = \frac{1}{\exp\left(c_a \theta f\right)} \cdot (1 + \frac{0.7}{L_a})^{-0.2} \cdot \frac{1}{A}. \tag{2.7}$$

The value of standard CSF (i.e. the $\frac{1}{A}$ element) can be obtained using data from studies on contrast. It was measured by the authors themselves, which made it accurate and well-fitted for a used display. The contrast sensitivity was measured through the experimental procedure for four bands from the Laplacian pyramid with their central values equal to 4, 8, 16, and 32 cycles per degree. The values of contrast sensitivity at those points (s_4, s_8, s_{16}, and s_{32}) were interpolated using cubic Hermite spline so that the contrast sensitivity could be calculated for any given spatial frequency in the image.

Transducer Model for Contrast Perception The last perceptual phenomenon employed in the model is *visual masking*. Its definition is a reduction of visibility of a specific image area due to the presence of an additional signal [8]. Such a phenomenon occurs in the individual spatial frequency channels, each responsible for only a single band of spatial frequencies. Visual masking divides into two basic types: self-masking and spatial masking [90].

Self-masking becomes apparent in the neuron structure which registers fre-

quencies inside the brain. Every spatial frequency channel quantizes image signal in the nonuniform manner. Because of that, the self-masking transducer has to take a non-linear form. Zeng et al. [90] defines it in such a way:

$$y(f,p) = \text{sign}(C_n(f,p)) \cdot |C_n(f,p)|^{\alpha}, \tag{2.8}$$

where α is a normalization factor ($\alpha \in [0,1]$).

Spatial masking considers contrasts of pixels lying in the neighbourhood of the analysed pixel. It is used to normalize self-masking contrast in order to account for nonuniform background masking. It is given with the formula:

$$w(f,p) = 1 + \frac{1}{|N|} \sum_{q \in N(p)} |C_n(f,q)|^{\beta}, \tag{2.9}$$

where N is the area around the pixel p, and β is the level of neighbourhood significance on the masking value ($\beta \in [0,1]$). Set $N(p)$ contains all points close to the pixel p. In this work, the size of this region was set to 5×5.

Since the value of spatial masking is used for normalisation, the final value of perceptual contrast is the quotient of 2.8 and 2.9 equations results:

$$C_t(f,p) = \frac{\text{sign}(C_n(f,p)) \cdot |C_n(f,p)|^{\alpha}}{1 + \frac{1}{|N|} \sum_{q \in N(p)} |C_n(f,q)|^{\beta}}. \tag{2.10}$$

This way, the calculation of perceptual contrast is finalised for every pixel of every spatial frequency band. The next stage of the predictor is the transformation of those values into the measure of possible resolution reduction.

2.2.3 Acceptable resolution reduction

The calculated perceptual contrast is defined in *just-noticeable difference* (JND) units. The difference between the contrast of two fragments is equal to one when that difference is detectable in 50% of cases and undetectable in the rest. The acceptable resolution reduction can be expressed as convolving the image with a Gaussian kernel with a given standard deviation σ_s value. The goal is to blur the image so that it is not distinguishable from the original by maximising the σ_s for the following inequality:

$$\forall_{p \in \Pi, f} \; C_t(f,p) - C'_t(f,p) \leq 1, \tag{2.11}$$

where Π is the analysed image patch and $C'_t(p, f)$ is the perceptual contrast of the blurred image with σ_s parameter.

Let G_{σ_s} be the Gaussian filter applied to the original image patch in order to blur it, given by a formula:

$$G_{\sigma_s}(x) = \frac{1}{\sqrt{2\pi}\sigma_s} \cdot \exp \frac{-x^2}{2\sigma_s^2}. \tag{2.12}$$

Dividing an image into Laplacian layers results in the alteration of frequency values in the blurred image in comparison to the original. This change of frequency is modelled by the following equation:

$$C'(f, p) = \hat{G}_{\sigma_s}(f) \cdot C(f, p), \tag{2.13}$$

where $\hat{G}_{\sigma_s}(f)$ is a frequency response of a Gaussian filter in a pyramid layer. Frequency response can also be calculated based on the Gaussian filter definition from 2.12. By transforming into frequency domain using Fourier transform the following formula is obtained:

$$\hat{G}_{\sigma_s}(f) = \exp \frac{-f^2}{2\sigma_f^2}, \tag{2.14}$$

where σ_f is standard deviation in frequency domain, calculated the following way:

$$\sigma_s \cdot \sigma_f = \frac{1}{2\pi}. \tag{2.15}$$

The combinations of 2.13 and 2.14 results in:

$$\sigma_s = \frac{f}{\sqrt{-2 \ln \frac{C'(f,p)}{C(f,p)}}} = \frac{f}{\sqrt{-2 \ln \frac{C'(f,p) \cdot S_{CSF}(f,\theta(p),L_a(f,p))}{C(f,p) \cdot S_{CSF}(f,\theta(p),L_a(f,p))}}} \overset{(2.3)}{=} \frac{f}{\sqrt{-2 \ln \frac{C'_n(f,p)}{C_n(f,p)}}}. \tag{2.16}$$

Because $C_n(f, p)$ may be calculated using the predictor executed on the original patch, the only unknown needed to calculate σ_s is $C'_n(f, p)$. Its value can be obtained using Equation 2.11. The σ_s value has to be maximised, so the difference between $C'(f, p)$ and $C(f, p)$ increases with the higher blur value. Therefore, the inequality

can be transformed into the equality:

$$C_t(f,p) - C'_t(f,p) = 1. \tag{2.17}$$

Next, using the values from the masking equation 2.10:

$$\frac{\text{sign}(C_n(f,p)) \cdot |C_n(f,p)|^\alpha}{1 + \frac{1}{|N|}\sum_{q\in N(p)} |C_n(f,q)|^\beta} - \frac{\text{sign}(C'_n(f,p)) \cdot |C'_n(f,p)|^\alpha}{1 + \frac{1}{|N|}\sum_{q\in N(p)} |C'_n(f,q)|^\beta} = 1 \tag{2.18}$$

The left side of the equation cannot be reduced to the common denominator in any simple way. Therefore, the following statement is assumed: spatial masking (the denominator of both parts of the summation) reaches similar values for both $C'(f,p)$ and $C(f,p)$. This assumption is based on the fact that uniform blurring of pixels lying in the neighbourhood of the analysed point does not change the spatial masking effect. Because of that, the equation can be simplified to the following formula:

$$\frac{\text{sign}(C_n(f,p)) \cdot |C_n(f,p)|^\alpha - \text{sign}(C'_n(f,p)) \cdot |C'_n(f,p)|^\alpha}{1 + \frac{1}{|N|}\sum_{q\in N(p)} |C_n(f,q)|^\beta} = 1. \tag{2.19}$$

Since the goal is to obtain the magnitude of contrast only, the signs of the operations can be omitted. As a consequence, the above formula derives into:

$$C_n(f,p)' = \left| |C_n(f,p)|^\alpha - (1 + \frac{1}{|N|}\sum_{q\in N(p)} |C_n(f,q)|^\beta) \right|^{\frac{1}{\alpha}} \tag{2.20}$$

Thus the needed σ_f value can be obtained.

σ_s is calculated using gathered values for every pixel and frequency. However, only one value is needed for each patch (the reason for that is clarified in Section 2.2.4). To this end, the maximum for all of the σ_f values in all pyramid layers is taken. This way follows a conservative approach of ensuring they are indistinguishable in all layers. Next, a similar approach is carried out to get the maximum values in the whole $N \times N$ patch. Based on the performed experiments, the best results were obtained using the smooth max function:

$$\hat{\sigma}_f = \sum_{p\in\Pi} \frac{\sigma_f(p) \cdot \exp(\omega \cdot \sigma_f(p)}{\exp(\omega \cdot \sigma_f(p)}, \tag{2.21}$$

where ω is the smoothing parameter, scaling the function. Setting it to any non-negative real value allows us to obtain the average (for $\omega = 0$), maximum (for $\omega \to \infty$), and any other value in between, depending on the ω value.

As the last step the σ_f value is converted for each patch into σ_s, according to the equation 2.15.

2.2.4 Experimental evaluation

Predictor explained in the previous subsection can be directly included in the graphic pipeline to implement the content-dependent foveated rendering. It does however have a list of parameters which have to be obtained beforehand. These are:

- self-masking and spatial-masking parameters (α, β),
- CSF parameters (s_4, s_8, s_{16}, s_{32}),
- fundamental eccentricity parameter (c_a),
- smooth max parameter (ω).

Therefore, a calibration procedure must be completed first to acquire the optimal values of all the parameters. Next, the experiments verifying the thesis can be performed.

To correctly set the model parameters, the capabilities of human vision for noticing the foveation must be tested. To this end, an experiment was evaluated, where the detection of blur was tested.

Evaluation method In the general case, this calibration study would require considerable time. The blur detection would have to be tested in all image regions by including the gaze position information. The images used for calibration would have to include all possible stimuli to consider virtually any rendering situation. In this project such a procedure was modified by testing multiple blurring levels simultaneously. Such an approach significantly increases the speed of the experimentation process.

For the full resolution display nonuniform blur is used. Near the centre of the screen the blur is the lowest (i.e. the σ of the Gaussian kernel is the lowest) and it increases with eccentricity. The formula for calculating the σ_s parameter is as follows:

$$\sigma_s(\theta) = \begin{cases} 0, & \text{if } \theta < r, \\ k \cdot (\theta - r), & \text{if } \theta \geq r, \end{cases} \tag{2.22}$$

where θ is an eccentricity calculated from the centre of the screen in degrees, r parameter marks the size of the centre region without any blur. k is the slope of the line starting right after the no-blur zone. For every tested image three different values are applied for both parameters (so 9 in total): $r \in \{4, 7, 11\}$ and $k \in \{0.0017, 0.0035, 0.0052\}$.

Stimuli images It is impossible to anticipate every possible perceptual effect, so only the most significant images were used for the studies. Thirty-six patches of 128×128 size were used. Eighteen of them have been taken directly from the Describable Texture Dataset set [13]. According to the authors, it contains all the most profound patterns visible in nature.

The remaining eighteen images were selected from an additional set of images containing 5640 photos. A greedy algorithm was used for selection to maximise the difference between any given patches. The choice of the next patch depended on the formula:

$$I_{n+1} = \arg\max_{I} \{d(I, \mathcal{D}_n)\}, \tag{2.23}$$

where $\mathcal{D}_n$ is a set of already obtained patches, I_{n+1} is the new patch added to the set, and $d(I, \mathcal{D}_n)$ is a function calculating the distance between two patches. It is computed based on the Laplacian pyramid, given by an equation:

$$d(I, \mathcal{D}_n) = \sum_{k=1}^{K} \left| \mathcal{L}_k(I) - \frac{1}{|\mathcal{D}_n|} \sum_{I_d \in \mathcal{D}_n} \mathcal{L}_k(I_d) \right|, \tag{2.24}$$

where K is the number of analysed pyramid layers and $\mathcal{L}_k(I)$ is the mean absolute deviation of all the pixels in the layer. In each iteration, the set was expanding until it reached a total of 18 photos. Thanks to this strategy, all the chosen images have unique frequency features.

Experimental procedure The overall number of images displayed during a testing procedure is 324 (9×36). Every single test has been repeated ten times for every participant. Such a test consisted of displaying the foveated and non-foveated images, between which users could switch freely. Their task was to choose which of the displayed images was the original, non-blurred one. The protocol followed the 2AFC procedure. A Tobii Eye Tracker was used, which forced the user to look at the centre of the screen. If the user moved their gaze outside the centre, the screen

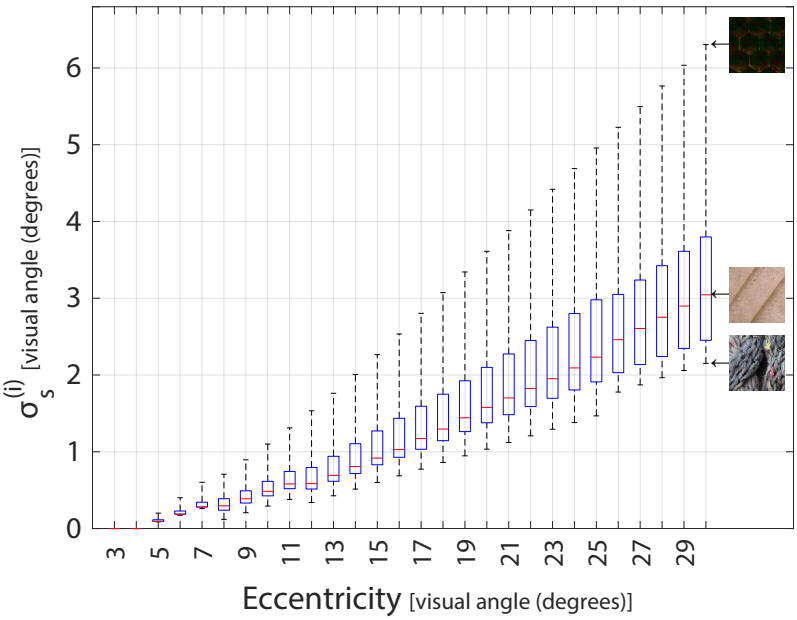


Figure 2.3: The results from the calibration experiment. Red line marks the median of σ_s parameters across all images. Blue boxes represent the middle 50% of all data points. Dotted line extends to the data points for images with lowest and highest detection rates (reused image from own publication [71]).

would have turned grey. It would display an image again after the user moved their gaze back to the centre. Additionally, the grey screen was used as a transition phase if the user switched from foveated to the original image (and vice versa).

Because the set contains only tiny patches, they have been tiled across the screen. The subsequent patches were mirrored to create a common border between themselves. This way, the sudden changes in the frequencies in the areas between the patches were avoided.

Because of its length, the whole experimental procedure has been divided into six sessions, each taking approximately 40 minutes. In total, it was completed by eight people.

Results

The results were obtained by counting the correctly classified foveated images as (r, k, θ) triplets. If the correctly classified ratio to all images was higher than 0.75, the stimuli were classified as undetectable. Therefore, for each undetected patch (with a detection factor higher than 0.75), the maximal invisible σ_s value can be calculated for every θ eccentricity. By averaging the data across all observers, the final results are obtained, as shown in Figure 2.3.

Collected data is sufficient to acquire model parameters. It was computed using Adaptive Simulated Annealing optimization [1]. It finds such parameters that

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

Table 2.1: The parameters obtained from calibration procedure.

the calculated predictor value $\hat{\sigma}_s$ is as close as possible to the σ_s acquired during the experiment. After finishing the optimisation procedure, the parameters were fine-tuned further using the gradient descent minimisation. The minimising function was defined as follows:

$$E = \min_{\mathbb{S}} \frac{1}{36} \sum_{i=1}^{36} \sum_{\theta=4^\circ}^{30^\circ} w_1(\theta) \left| w_2(\hat{\sigma}_s^{(i)}(\theta) - \sigma_s^{(i)}(\theta)) \right|, \tag{2.25}$$

where $\mathbb{S}$ is the set of all calibrated parameters (α, β, s_4, s_8, s_{16}, s_{32}, c_a, and ω), $\sigma_s^{(i)}$ is a ground truth value from the experiment for image i, $\hat{\sigma}_s^{(i)}$ is value computed using predictor, and (w_1, w_2) are additional used weights. w_1 is defined as follows:

$$w_1(\theta) = \begin{cases} 2, & \text{if } \theta < 10, \\ 1, & \text{otherwise.} \end{cases} \tag{2.26}$$

By including this weight, the algorithm is adjusted to shift the higher precision into the near foveal region of the image. The purpose of weight w_2 is to penalize the higher estimated values more than lower in order to omit the potential lack of important features. It is given using the formula:

$$w_2(x) = \begin{cases} 8x, & \text{if } x > 0, \\ x, & \text{otherwise.} \end{cases} \tag{2.27}$$

Such an optimisation procedure results in the final values of the parameters presented in Table 2.1.

These results conclude the calibration of the predictor. The foveation procedure can be included in the rendering pipeline with all the necessary data.

Performance tests For this project, several different versions of the predictor were implemented. The first implementation was created to test the performance of the additional computations required to estimate acceptable blur. The implementation was made using C++ and the OpenGL API. For this purpose, all the mechanisms to increase the efficiency of the calculations are used:

- the image for predictor calculation has such a size that it represents 1/16 of the original image,
- every single predictor step performed on the entire image was implemented with a shader that processes the pixels in every layer in parallel,
- the separation of the image into layers of the Laplacian and Gaussian pyramids was performed using mipmapping.

For the used 2560×1440 display, the input size of the predictor was set to 128×128. Such a reduction in dimension size relative to the full-resolution image allowed for a significant increase in performance: the time of calculating the predictor for the 2560×1440 image using the NVIDIA GeForce 2080 Ti graphics card is 3.0 ms and for the 128×128 image – 0.7 ms.

This implementation was extended with rendering to test the entire system's performance. The predictor was connected to the OpenGL rendering system that had initially synthesised the scene and displayed the result on the screen. The synthesis was performed only on the reduced image (the previously mentioned size of 128×128). The obtained $\hat{\sigma}_s$ values can be converted into the actual resolution change using the Variable Rate Shading (VRS) [55] technology.

VRS technology enables specifying the sample rate in 16×16 patches of the image. Its possible settings are 1/1, 1/2, 1/4, 1/8, 1/16 of total samples. The 1/2 and 1/8 samplings were rejected, as their inclusion would lead to areas with inhomogeneous sampling, which were not considered when designing the predictor.

The value of the $\hat{\sigma}_s$ parameter has to be converted to a sampling value to obtain the sampling level in a given area of 16×16 pixels. Due to the theoretically infinite Gaussian filter window size needed to blur the image, the sampling shift can be estimated only. Therefore, only the 95% area corresponding to the blurred value, i.e. the area where the distance from the centre to its sides is $2\hat{\sigma}_s$, was taken into account. The minimum sampling rate concerning $\hat{\sigma}_s$ is calculated using the Nyquist rate definition as:

$$S_{1D} = \frac{1}{4\hat{\sigma}_s}, \tag{2.28}$$

Figure 2.4: Images used for benchmarking the methods (reused image from own publication [71]).

Scene	Full resolution	Content-aware foveation	Standard foveation
Sponza	2.6 ms	3.0 ms	2.3 ms
Water	9.5 ms	4.3 ms	5.3 ms
Fog	22.9 ms	5.5 ms	13.9 ms

Table 2.2: The rendering time for various scenes and techniques.

where S_{1D} is the sampling rate in the one-dimensional case. The two-dimensional version of the equation (which was used to sample the image) is as follows:

$$S_{2D} = \frac{1}{16\hat{\sigma}_s^2}. \tag{2.29}$$

The technique's performance was tested in three scenarios: rendering with a Phong-shaded Sponza scene, water simulation with reflections, and fog created through volumetric effects. Each type of rendering has been evaluated using standard rendering, content-aware predictor rendering, and standard foveation rendering without using the predictor. The rendered images are shown in Figure 2.4 and the performance results in Table 2.2.

The content-aware method is more efficient than the other methods for rendering water and fog. Simple rendering using Phong shaders has a lower performance score. It is caused by the fact that calculating the predictor takes a fixed amount of time, which has a higher impact the shorter the total rendering time is. However, in this scene the rendering speed is high enough to make the foveation unnecessary. The technological limitations must also be considered. The primary limitation of the VRS technique is the minimum sampling rate limit, which is 1/16. For this reason, the performance results do not fully reflect the system's capabilities.

Model validation In order to verify the accuracy of the predictor, a series of experiments were conducted. To this end, another implementation was prepared using the Unity3D environment. The blur was used to lower the image resolution, making the obtained images entirely consistent with the model. The predictor results for the sample images are shown in Figure 2.5. The picture consists of three horizontal

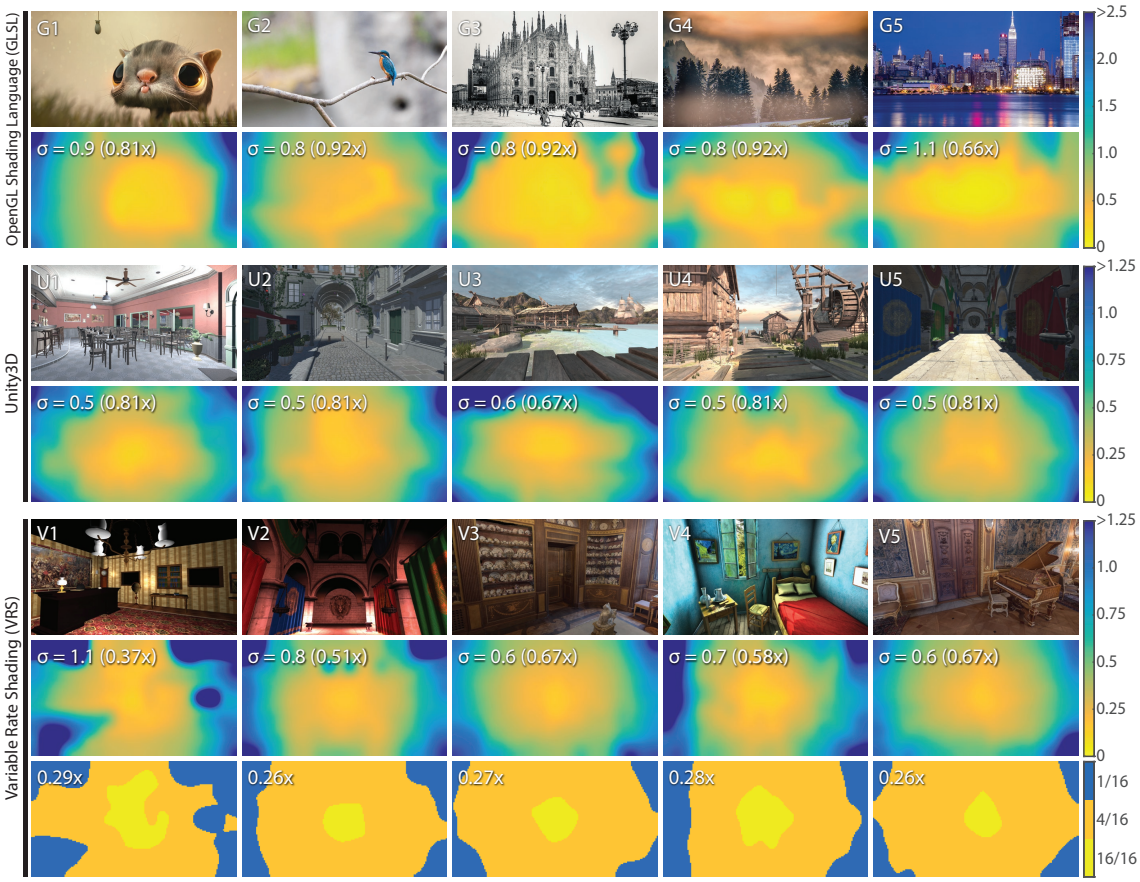


Figure 2.5: Sample images captured from the predictor implementations and the sampling maps, with the assumption that the gaze position is at the centre of the image. The numbers in the top left corners of maps show the average $\hat{\sigma}_s$ of the frame. Numbers in parenthesis indicate the required number of samples relative to the standard foveation. The shading map used to render VRS is shown in the last row. The number in the top left corner indicated the fraction of samples rendered regarding the full resolution (reused image from own publication [71]).

sections. The first and second each have two rows and show the displayed image and the $\hat{\sigma}_s$ map computed for it. In the third Section, a pixel grid is shown as well. It determines the actual reduction of sampling through the VRS.

In the first experiment, content-aware foveation was tested to determine whether it was detectable or not. To this end, the participants were shown the rendering with foveation and compared it to the rendering without it. The participants did not know which image was the foveated one. They were asked to select the one that looked better. In order to confirm the blur coefficients from the experiments, an additional experiment version was tested, in which the $\hat{\sigma}_s$ parameters were multiplied by a constant (0.5-5), i.e. the resolution loss was greater or lower than the calculated values.

All implementations were tested for this experiment: GLSL with static im-

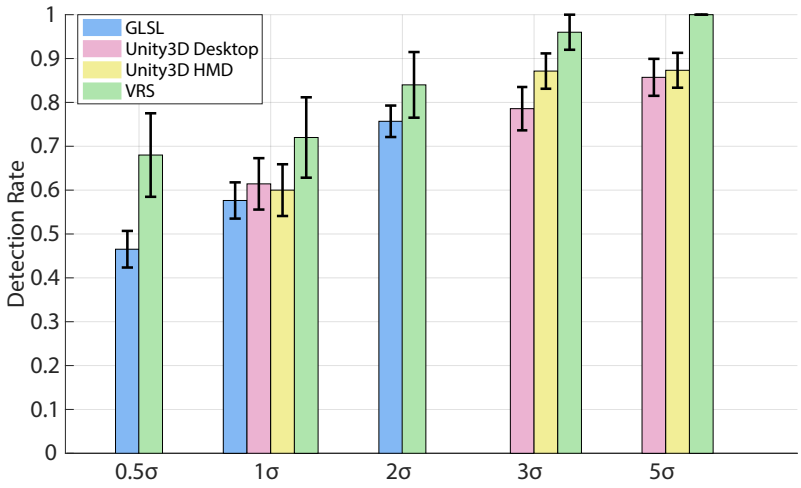


Figure 2.6: The results of the experiments computing the detection rate of foveation at different levels of resolution reduction and using different implementations. The standard deviation is shown with black whiskers over each bar (reused image from own publication [71]).

ages only, Unity3D displayed on a standard display, Unity3D with a Virtual Reality headset, and GLSL with VRS technology for rendering dynamic scenes. For all of those implementations, the Tobii Eye Tracker hardware was used. That made it possible to gather the gaze position and use it in the prediction. For GLSL, Unity3D, and VRS experiments, 10, 12 and 5 participants took part, accordingly. They were shown multiple scenes in random order with repetitions. The results of this experiment are presented in Figure 2.6.

The results show that all implementations keep the detection level below 0.75 (1 JND) if the σ of Gaussian blur is at or below calculated $\hat{\sigma}_s$. Increasing the σ parameter increases the detection rate, confirming the model. It can also be observed that the detection rate using VRS is significantly higher than for other methods. The cause of such a phenomenon is visible temporal artefacts created due to the actual decrease in sampling rate. Even though it is not accounted for, the results are still below the JND threshold value.

In the second experiment, the method was compared to the standard foveation, in which only the eccentricity information was considered to compute the acceptable resolution decrease. To make the comparison fair, the resolution reduction was adjusted in standard foveation so that, in total, it was equal to that of the content-aware method. The experiments were divided into two parts. In the first part, the participant was shown standard foveation with three different foveal region radii ($r \in \{4, 7, 11\}$) and was asked to select the one which looked best. The foveal region was displayed continually in full resolution, and the rest of the image had linearly increasing resolution reduction. The slope of this linear increase was

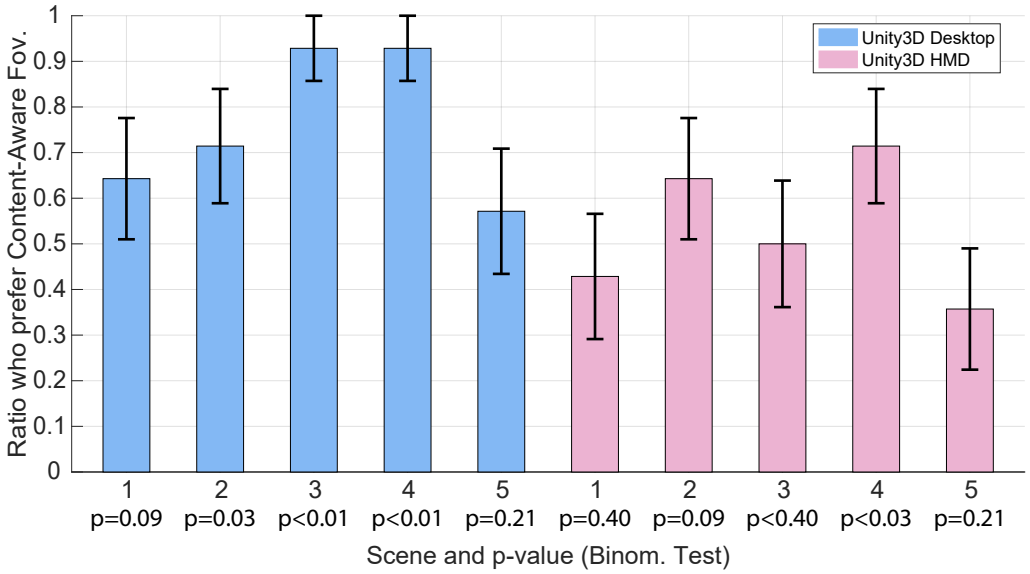


Figure 2.7: The results of the experiment comparing standard foveation with content-aware foveation. Values above 0.5 mean a higher preference for the content-aware method compared to the standard. The numbers directly below the bars are the references to the scene numbers, and they correspond to scenes U1-U5 from Figure 2.5. The p-values are the results of the binomial test. The standard deviation is shown as black whiskers over each bar (reused image from own publication [71]).

computed so that the total resolution reduction was the same as the content-aware foveation model. In the second part, the participant had to select between the rendering with standard foveation and the predictor. The procedure was similar to the first experiment regarding randomness and stimuli selection. It was performed on 14 participants in the Unity3D implementation for VR headsets and standard display on five scenes. The results from this experiment are shown in Figure 2.7.

The results show that the participants preferred the content-aware method in the desktop version. The participants selected it 53 out of 70 times in total. The results are statistically significant ($p < 0.01$). The content-aware method was selected in 37 out of 70 trials on the VR headset. The results are not statistically significant ($p = 0.28$). This result is caused presumebly by the limited resolution of VR and lens distortions that were not considered.

2.3 Visual adaptation for foveated rendering

Directional vision employed by foveated rendering is also a subject of visual adaptation. Due to rapid changes in gaze location, the human visual system continuously adapts to the newly observed pixels. Such a trait can be considered maladaptation, as it is impossible to be fully adapted to the non-uniformly lit environment (see Section 1.2). Since this phenomenon plays a large role in colour and contrast perception (see Section 1.2), it is cogent to include it in the foveated rendering. In this Section, a visual adaptation model is proposed. It does not improve the performance of the rendering process, but its design is based on the changes in visual acuity caused by nonuniform photoreceptors arrangement.

2.3.1 Background

Visual adaptation splits into two main phenomena: bright and dark adaptation. Adaptation to brightness takes place due to the increase of the adaptation luminance. After leaving the darkness, rods are adjusted to low luminance values and remain at high sensitivity. A drastic increase in brightness bleaches them and causes pain [15]. The vision for a moment has a very high perceived brightness. It quickly comes back to normal after the cones readjust their sensitivity.

The opposite phenomenon happens while entering the darkness from a well-lit environment. At first, the vision is entirely black, as the sensitivity of rods and cones is very low. A bright environment bleached the rods. With time passing, they start to increase their sensitivity slowly. Before they can recover, the cones increase their sensitivity until they reach the maximum. After that, the sensitivity of rods slowly increases, and more and more details of the environment become visible. Compared to bright adaptation, adjusting to darkness typically takes a long time. Depending on the amount of available light and the initial rod's sensitivity, this process might take from 10 minutes up to 2 hours. A plot of threshold luminance (minimum detectable luminance) is presented as a function of time in Figure 2.8.

Typical computer monitors or VR goggles cannot display a wide range of luminances. Therefore, the actual adaptation to the rendered content is minuscule. However, the rendered scene can be treated as a world model with actual luminance values. Even if the displayed sun can only reach $300\frac{cd}{m^2}$ luminance on the screen, it can be modelled as the accurate $10^8\frac{cd}{m^2}$. All colours can be correctly presented according to any given adaptation luminance using tone mapping [77]. The temporal aspect can also be simulated using the luminance threshold function.

Due to the nature of computer games and typical VR applications, slow and detailed adaptation might not be what users desire. In this project, multiple

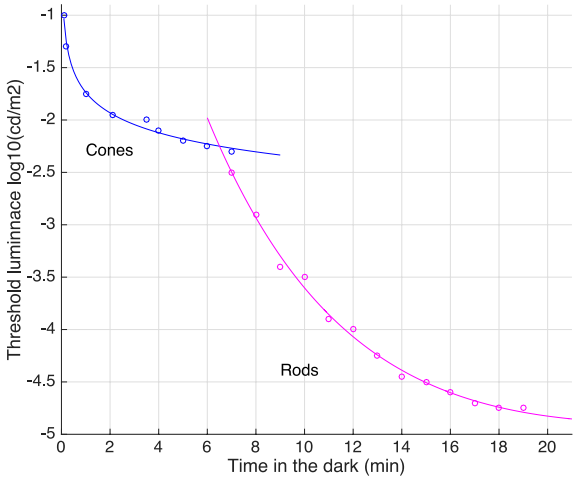


Figure 2.8: The threshold luminance as a function of time during adaptation to darkness. Blue lines represent the values which were reached thanks to the cones. Pink values become available after the sensitivity of rods increases beyond that of cones. Data points after [39] (reused image from own publication [80]).

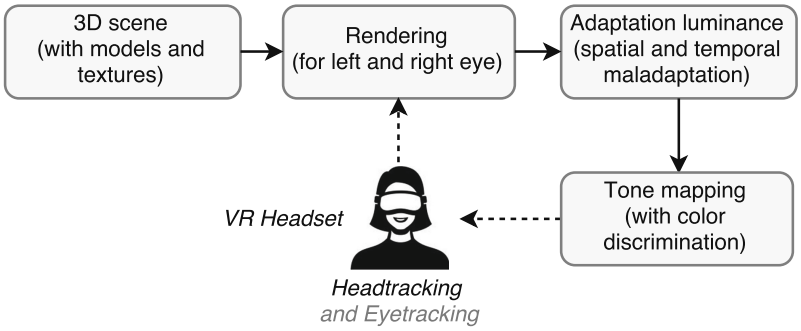


Figure 2.9: A scheme of the visual adaptation architecture (reused image from own publication [80]).

adaptation models were created and compared. They were evaluated through the experimental procedure.

2.3.2 System architecture

The visual adaptation model is divided into several stages, as shown in Figure 2.9. Firstly, the scene is fully rendered to a floating-point texture based on the scene description. Then the adaptation luminance is calculated. It considers the user's gaze location, the luminance of the observed environment, and the luminance to which the observer is currently adapted. The details concerning this part of the model are given in Section 2.3.4. Next, calculated adaptation luminance is used to map the tones onto the used display range. The mapping is done following the human visual system, including the appropriate colour discrimination (see Section 1.1). This process is described in Section 2.3.3. Finally, the image is presented to the observer.

2.3.3 Tone mapping

The adaptation framework consists of a renderer and a post-processing effect set. The lighting formulas were computed using photometric space in the shaders. Doing so allows capturing correct luminance values of the simulated world and applying the adaptation simulation. Five scenes were used in total, as shown in Figure 2.11.

Display limitations make it impossible to render images at appropriate luminance levels. Instead, they are transformed into the RGB range specific to the display in this work. To this end, a tone mapping is performed based on the maximal luminance value of the display and human visual system characteristics [61].

Ward et al. [77] defines a tone reproduction operator as:

$$L_d = m \cdot L_w, \tag{2.30}$$

where L_w is the simulated world luminance, L_d is the mapped display luminance, and m is a scaling multiplier. m parameter can be calculated using threshold-versus-intensity function (t.v.i.) [77]. Ward estimates it as:

$$m(L_{da}, L_{wa}) = \frac{t(L_{da})}{t(L_{wa})}, \tag{2.31}$$

where L_{da} and L_{wa} are adaptation luminances of display and world observer, accordingly, and $t(L)$ is a t.v.i. function value for luminance L.

To calculate the t.v.i. function for both display and world observers, the idea of Ferwerda et al. [18] is employed to model cones and rods separately. Rods t.v.i. for scotopic vision is:

$$\log t_s(L_a) = \begin{cases} -2.86, & \text{if } \log L_a \leq -3.94, \\ \log L_a - 0.395, & \text{if } \log L_a \geq -1.44, \\ (0.405 \log L_a + 1.6)^{2.18} - 0.72, & \text{otherwise.} \end{cases} \tag{2.32}$$

Cones t.v.i. for photopic:

$$\log t_p(L_a) = \begin{cases} -0.72, & \text{if } \log L_a \leq -2.6, \\ \log L_a - 1.255, & \text{if } \log L_a \geq 1.9, \\ (0.249 \log L_a + 0.65)^{2.7} - 0.72, & \text{otherwise.} \end{cases} \tag{2.33}$$

The above equations in dark and bright environments estimate scotopic and photopic vision. The *mesopic vision* is also considered, which takes place in between. It ranges from $0.03 cd/m^2$ to $3.00 cd/m^2$ [27].

To model all vision states at once, the photopic and scotopic models were combined into one:

$$t(L_a) = (1 - k(L_a)) \cdot t_p(L_a) + k(L_a) \cdot t_s(L_a), \tag{2.34}$$

where k is a function that has values between 0 (photopic) and 1 (scotopic). Anything not equal to 0 or 1 is considered mesopic vision.

The adaptation luminance cannot be measured for display observers. Therefore, the display is assumed to be the only light source, and the user adapts to its maximal luminance value. For the world observer, content and gaze direction influence the adaptation luminance the most. Gaze-dependent contrast sensitivity function was used [60] to measure the world observer's adaptation luminance. Its values depend on the number of cones in the retina and their distribution, which directly influences the adaptation luminance [51].

In every frame, a map with dimensions equal to the dimensions of the rendered image is generated. Every pixel is computed as a spatial cutoff frequency function value depending on its location and current gaze position. Then the map is used to weigh the pixels of the rendered frame to compute weighted adaptation luminance.

After calculating the adaptation luminance for both world and display observers, one additional element of the visual system is modelled: colour discrimination. As mentioned previously, rods can create monochromatic vision only. As such, the final colour is calculated as the sum of rods and cones' outputs:

$$LDR_{RGB} = \sigma \cdot L_{da} + (1 - \sigma) \cdot \frac{HDR_{RGB}}{L_{wa}} \cdot L_d a, \tag{2.35}$$

where LDR_{RGB} is colour intensity mapped onto a display, and HDR_{RGB} is the original simulated world luminance. σ parameter controls the rods impact. It is defined using Hunt et al. [27] formula:

$$\sigma = \begin{cases} 1, & \text{if } L_{wa} < 0.03 cd/m^2, \\ 0, & \text{if } L_{wa} > 3.00 cd/m^2, \\ 0.07022/(0.06929 + 1.409 \cdot \exp(4.267 L_{wa})), & \text{otherwise.} \end{cases} \tag{2.36}$$

This function has a sigmoidal trend, as visualised in Figure 2.10.

Images at a completed adaptation state can be created using tone reproduction formulas and colour discrimination, assuming that the observer is fully adapted to it. The exemplar scenes presenting this status are shown in Figure 2.11. In the next Section, the temporal aspect of adaptation is modelled.

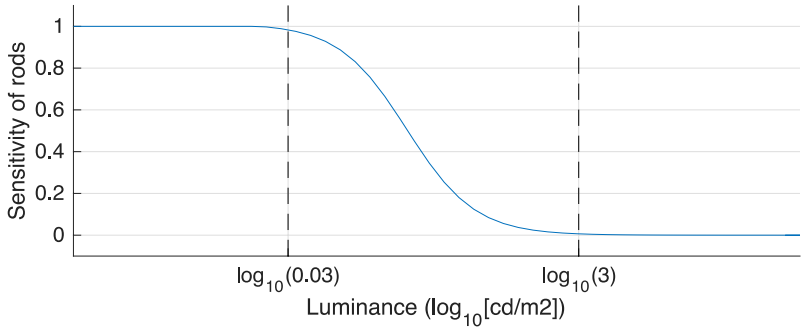


Figure 2.10: Function estimating rods sensitivity. Dashes lines represent the borders between scotopic and mesopic vision (left), and between mesopic and photopic (right) (reused image from own publication [80]).

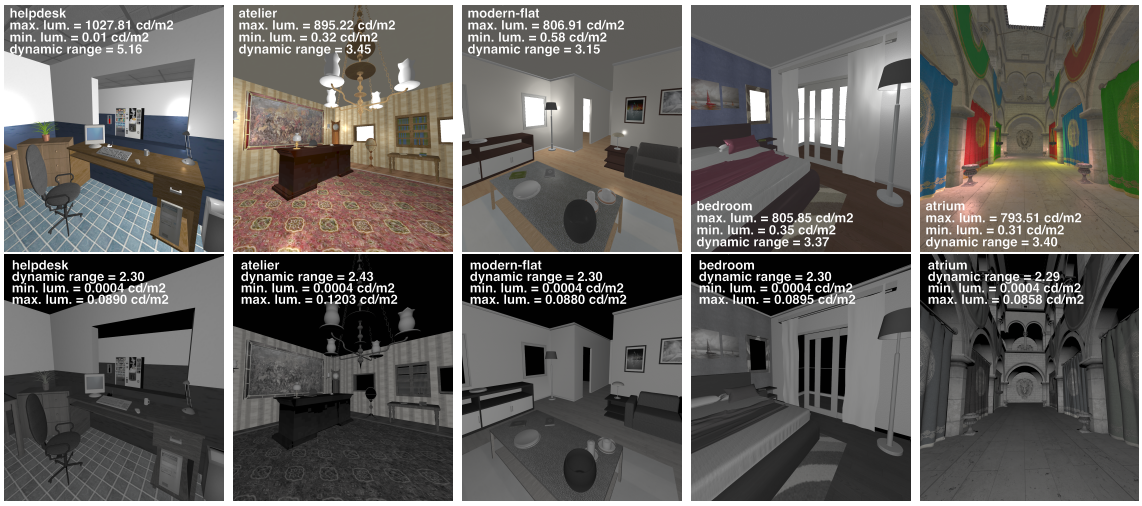


Figure 2.11: The rendering results of five exemplar scenes. Top row shows them illuminated by the light, bottom row is generated with lights turned off. Dynamic range is calculated as a logarithm of max luminance value in the image divided by logarithm of a minimum (reused image from own publication [80]).

2.3.4 Temporal effect

The adaptation to darkness takes place in two steps: cones adaptation, and rods adaptation, as shown in Figure 2.8. They are modelled by fitting the data points to the exponential functions:

$$\begin{aligned} \Delta L_{cone}(t) &= 5.659 \cdot t^{-0.051} - 7.431, \\ \Delta L_{rod}(t) &= -5.766 \cdot e^{-0.0053 \cdot t} + 9.694 \cdot e^{-0.1648 \cdot t}, \end{aligned} \tag{2.37}$$

where t is time in minutes. The combined threshold can be expressed as:

$$k(t) = \min(\log \Delta L_{cone}(t), \Delta L_{rod}(t)). \tag{2.38}$$

According to the Weber's law, the adaptation luminance decreases proportionally to the threshold value in the log-log space [39]. An inverted t.v.i. function is approximated in the following way:

$$\log L_a(t) = p_1 \cdot k(t) + p_2, \tag{2.39}$$

where $p_1 = 1.191$ and $p_2 = 0.7075$.

The accurate model of dark adaptation extends over a long period. Figure 2.8 demonstrates that it takes ca. 20 minutes for the threshold difference to become negligible. In fast-paced video games, such a long time would not be foreseeable. Therefore, the adaptation time was scaled down to 20 seconds to make it shorter.

Another aspect of dark adaptation that is not suitable for displays and VR headsets is the threshold value at the intersection point between cone and rod sensitivities. According to the model, this point lies below $0.005 cpd/m^2$. Such a low value would not be perceptible. Instead, this point was shifted by 1.75 log units upwards, causing the intersection to occur at the luminance of $0.3 cpd/m^2$. This shift distance was selected empirically.

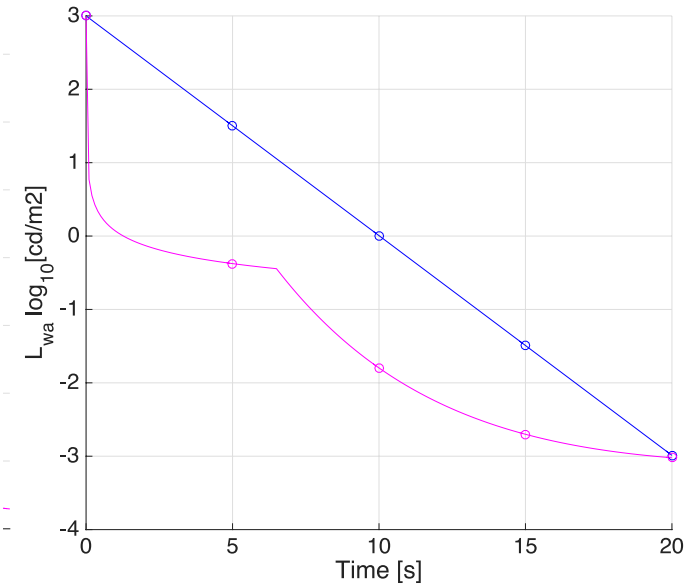


Figure 2.12: The threshold luminance curve after adjusting. Magenta line represents the perceptual adaptation, divided into cones and rods sensitivity changes. Blue lines shows the linear adaptation in log-space (reused image from own publication [80]).

The threshold luminance function was modified using the above changes. This model is referred to as *perceptual* because of its closeness to the human visual system. Another model was created, where the adaptation is not separated into cones and rods steps. This model is identified as *linear* due to its linear (in log

space) increase in the visibility threshold. Both the adaptation curves are presented in Figure 2.12. The differences between linear and perceptual adaptation are shown in Figure 2.13.

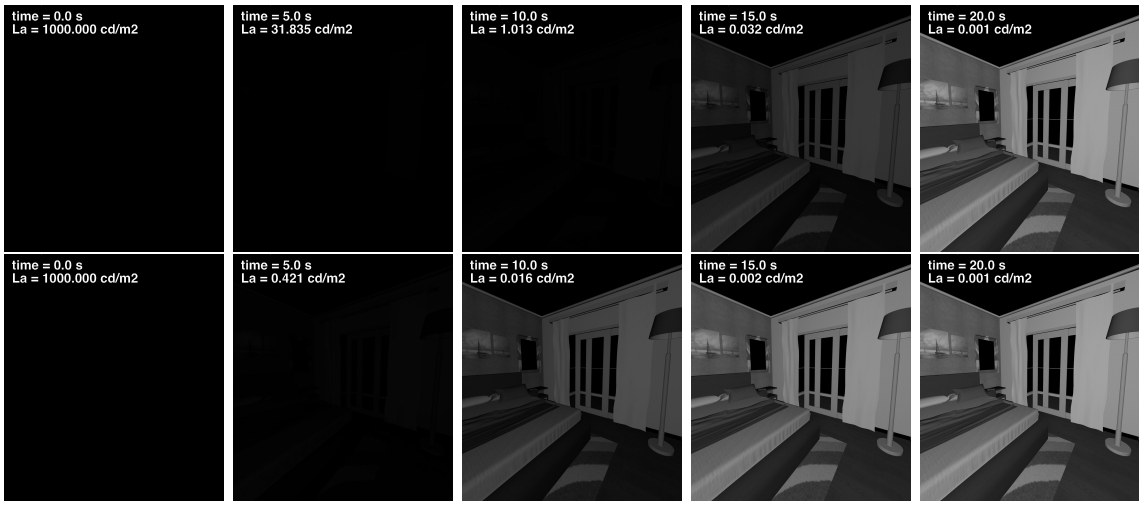


Figure 2.13: Exemplar process of dark adaptation for linear (top row) and perceptual model (bottom row). The world observer was initially adapted to bright environment with $L_a = 1000cd/m^2$. Then the light was turned off (reused image from own publication [80]).

2.3.5 Experimental evaluation and results

An experiment was created to test the preference for speed and type of adaptation. The participants were shown five scenes where the adaptation process was demonstrated. They were asked to look freely around in a well-lit environment. After six seconds, all the lights turned off, and the adaptation to darkness started. In the beginning, the scene was entirely dark, but its brightness and contrast gradually increased as time passed. The adaptation time was set to 5, 15, or 25 seconds. In a single step of the experiment, the participants were shown two of those speeds subsequently to emphasise the difference between them. Then, they were asked to select which speed they preferred. Every participant tested all possible combinations of timings for all scenes. The perceptual and linear adaptations were tested separately.

In the second experiment, the participants compared linear adaptation to perceptual. The adaptation time for both methods was fixed to 25 seconds.

Fifteen participants took part in the first experiment and nine in the second. The stimuli were shown in the VR setting.

The results of the first experiment are presented in Figure 2.14. The data shows that slower adaptation is more preferred for the linear model. The results are statistically significant and hold for all the tested scenes. However, results for

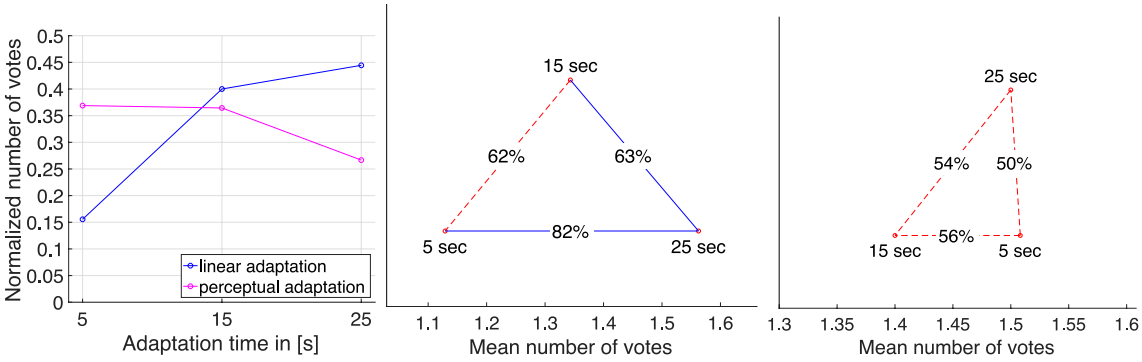


Figure 2.14: The results of the experiment estimating the preferred adaptation time. The left figure shows the fractions of all the favoured stimuli for all tested speeds. The central and right figures show the multiple-comparison test [52]. They represent the preference for all combinations of adaptation time for a linear (central) and perceptual (right) model. Numbers between the points represent the preference for selecting the right-most adaptation time. Solid lines represent statistically significant results (reused image from own publication [80]).

perceptual adaptation did now show statistical significance. This difference between adaptation speeds is indistinct due to a more complicated course of threshold changes. The changes in scene brightness and contrast most of the time are minor. The only rapid changes happen at the start (when cones regain their sensitivity) and after reaching the intersection point between rods and cones. These time frames are short enough to appear almost identical across all speeds.

The results of the second experiment showed that participants preferred the linear model over the perceptual with 68% of all votes ($p = 0.65$). The causes of such results are presumably the adjustments applied to the calculations – shortening the adaptation speed and shifting the visibility threshold values. Discarding those changes would not suit well in real-time virtual reality games. Therefore, the linear model fits better, as it is more visually pleasing.

2.4 Temporal coherence in near-eye displays

2.4.1 Objectives and previous work

The foveation rendering approaches suffer from the *temporal aliasing*. Its most apparent occurrence is visible at the object edges and areas with high spatial frequencies (see Figure 2.15). They can be solved using already established fast methods (e.g. SMAA, MSAA, MLAA, and FXAA [30, 29]). Unfortunately, for mentioned algorithms, temporal coherence is not guaranteed.

Current VR displays have a significantly lower spatial resolution because of their limited pixel count, closeness to the eye, and lens magnification. Compared

Figure 2.15: Two subsequently rendered frames with sub-pixel camera movement in-between. Zoomed areas show the changes in the pixel values caused by temporal aliasing.

to the standard display observed from the 60–70 cm distance, near-eye aliasing is significantly more pronounced. Additionally, due to constant head movements, the view from the cameras changes. As they disturb the vision to a large extent, extra measures are needed to improve the image quality.

This problem has been apparent since the early days of computer animation, as Korein et al. raised it in 1983 [41]. Switching between subsequent frames gathered at specific timeframes caused unnatural movements with jerkiness. The proposed smoothing of the images applies motion blur to simulate continuous transformations. This approach can be compared to the way physical cameras work. Each picture is captured over a specific period, controlled by shutter speed, averaging the motion.

The most recent works on the topic follow the idea of estimating motion. Yang et al. [86] shows a strategy of reusing samples from previous frames on the sub-pixel level for reprojection with a local spatial filter. This idea of temporal supersampling was further researched and resulted in multiple developments, such as TXAA and CPS [85, 83, 35].

All temporal anti-aliasing techniques introduce an additional cost. With modern GPUs, the cost scales from 0.5 to 7.0 ms, depending on the used technique [9]. Unfortunately, the use of foveated rendering increases the need for high-quality anti-aliasing. Decreased resolution in the peripheral vision increases the magnitude of aliasing artefacts which are still perceptible [70].

The perception of temporal aliasing depends heavily on eccentricity and content, e.g. the smoothness of objects, occlusions, post-processing, or lighting effects. Measuring the locations of perceptually disturbing aliasing beforehand could significantly reduce the performance hit. Unfortunately, there is no universal way of detecting such problems – modelling and accounting for all possible stimuli is not trivial. A convolutional neural network (CNN) is proposed to overcome this problem. A method of generating input for such a network is described, along with the

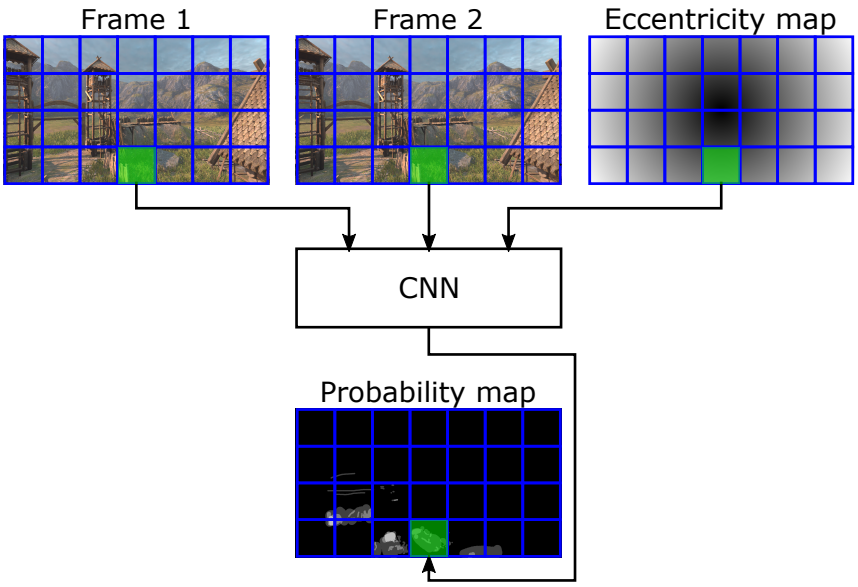


Figure 2.16: Overview of the aliasing detection procedure. Frame 1 and Frame 2 indicate two subsequently generated frames. CNN operates on patches and created a probability map as an output.

experimental procedure performed through the user study.

2.4.2 System architecture

The CNN was designed to estimate the probability of detecting aliasing in every pixel (see Figure 2.16). Two subsequent frames are needed to recognise the temporal artefacts. Therefore, the input to the network consists of two subsequent rendered frames with alleged aliasing in-between. To include the perceptual limitations in training, the network also received the *eccentricity map* that consists of the distances from the eye position to all pixels. As an output, the network provides the *probability map*. Values in the map indicate the probability of belonging to a specific aliasing class.

Network implementation The network's architecture is presented in Figure 2.17. The input (at the top) consists of two RGB frames and one single-channel eccentricity map. The data flows through a series of convolutions and pooling blocks. The network's output is the probability map generated through the softmax function.

Generating input for the network An experiment was prepared to gather the network's input and ground truth data. Short animations were recorded in a set of 3D scenes. The frames were captured using foveated rendering, assuming that the viewer is looking at the centre of the screen. The participants were shown two subsequent frames from those animations. They could switch between them freely

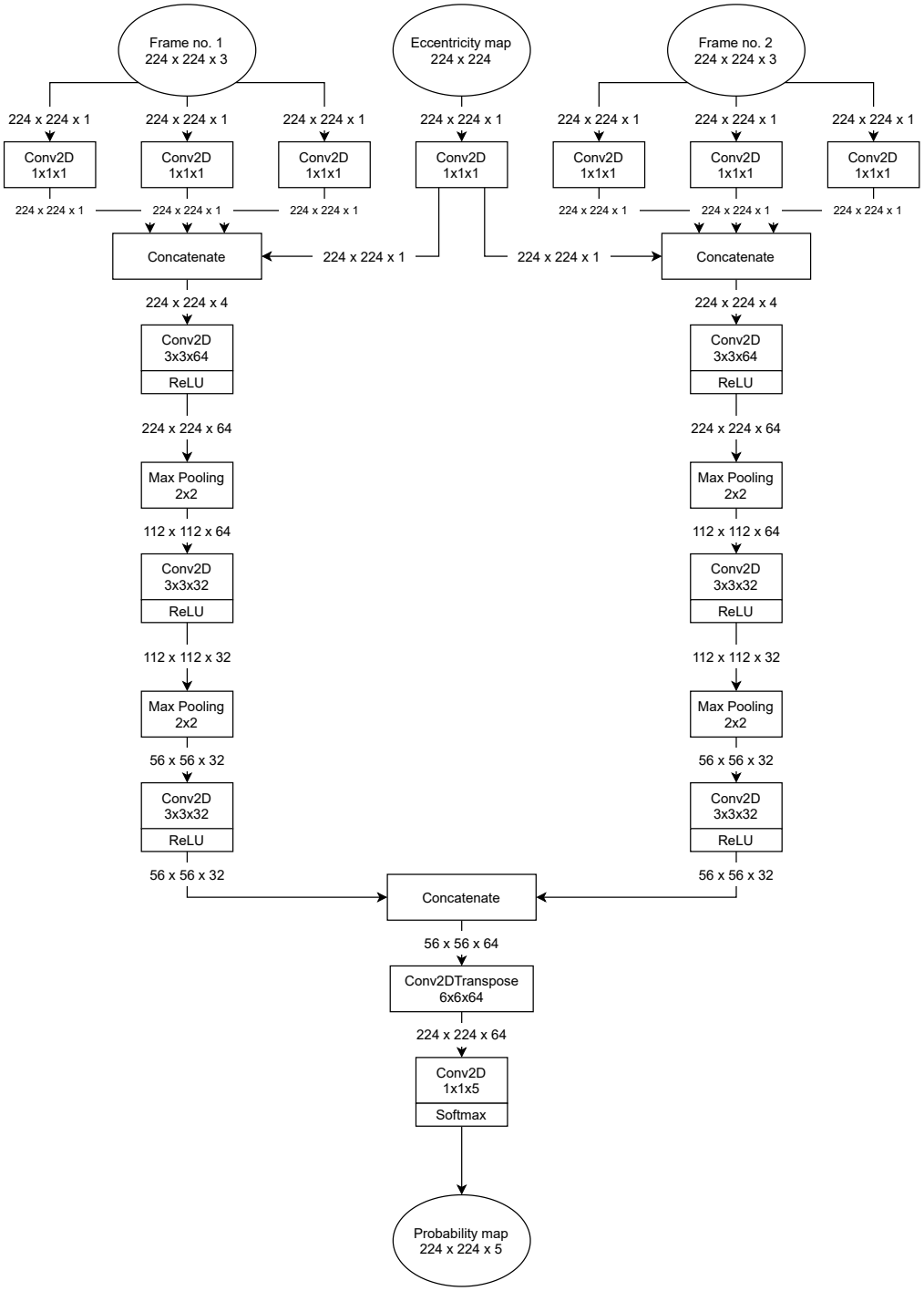


Figure 2.17: CNN architecture. Top of the figure shows the network's input, and bottom its output. Numbers in the convolution blocks indicate the kernel and filter size. Numbers between the blocks contain the size of the data.

Figure 2.18: Example frame from the animation and the aliasing map. The brightness of the pixel is scaled depending on the aliasing class.

during the experiment. Their task was to look at the centre and mark the visible distortions appearing while switching between frames.

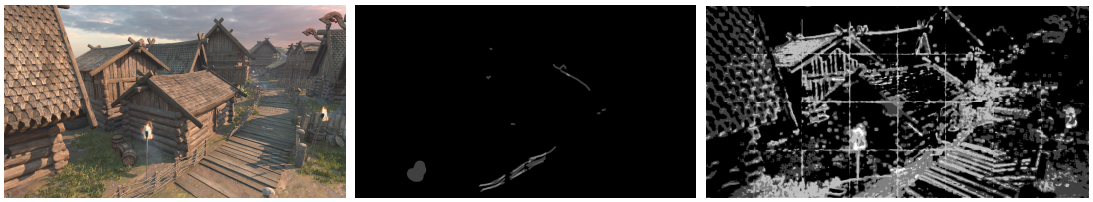
Ten scenes in total were prepared, with ten recorded animations for each. The dimensions of frames were 1920×1080. The experiment was conducted for four participants. Using the results, five classes of aliasing were defined:

- **Invisible** – pixels which were not marked by any of the participants,
- **Barely visible** – pixels marked by one participant only,
- **Just noticeable** – pixels marked by two participants,
- **Mostly detectable** – pixels marked by three participants,
- **Apparent** – pixels marked by all four participants.

An example of stimuli and a generated aliasing map are presented in Figure 2.18.

To generate input for the network, 224×224 patches were randomly extracted from the 1920×1080 images along with the eccentricity and ground truth probability map. To extend the stimuli range, 90°, 180°, and 270° rotations and mirroring were performed. 10k images were generated in total, which were then directly used for CNN training.

ADAM optimizer [37] was used for the training with 10^{-3} training rate. As a loss function, Categorical Cross-Entropy was applied. The weights of classes were



(a) Input frame (b) Ground truth (c) Final image

Figure 2.19: Presentation of training results. The final image is composed of 224 × 224 images taken from the output of the network.

adjusted to balance the loss values. The whole training procedure took 70 epochs until the loss function flattened out.

2.4.3 Results and discussion

The accuracy of the network was tested on the additional set of animations not used during the training. Generated patches were then connected to form a full resolution frame. An example result is presented in Figure 2.19.

The results show that the network infers considerably more pixels to contain aliasing artefacts than expected. Compared to the ground truth, significantly more edges and textures are marked as potentially temporally unstable. Additionally, the concatenation of patches to create a full image introduces border artefacts.

Mentioned problems are thought to have come from the limitations of the input data. The accuracy of the ground truth maps is low because of the limited number of participants. Additionally, 10K images come from 10 scenes only. Preparing a more extensive input set is a demanding task, as it requires a very diverse selection of environments. However, it would mitigate the training process for the network.

Despite its low accuracy, the network can limit the required number of anti-aliased samples. With an improved data set, the method may improve the overall accuracy and limit the anti-aliasing constraint further.

2.5 Chapter summary

Above studies have shown that including localised perception increases the performance and, compared to standard global foveation, increases the quality. Thanks to the analysis of the scene content, the sample count can be precisely measured for any stimulus. The implemented method is viable for both standard and near-eye displays. The gains might vary depending on the chosen rendering type (rasterisation or ray-tracing). The ray-tracing can be scaled by varying the number of

cast rays, so it benefits from such adjustments the most. The rasterisation cannot be thoroughly exhausted because there is no easy way to reduce any given pixel's resolution. This work relied on Nvidia VRS technology, which has its limitations in resolution reduction. Ray-tracing is gaining more ground in recent years, and VRS might either be improved or replaced with another variable sampling technology, which would prove this method to be more advantageous.

The foveated rendering system has been extended by including light adaptation, which relies on the gaze direction. The perceptual quality of the model was evaluated in various settings. The experiments have shown that typical users prefer slow and gradual adaptation compared with cones and rods modelled independently. The simplified linear model is easier to implement and computes faster, making it the best choice for adaptation simulation.

Besides spatial, the foveated rendering has been analysed in the temporal domain. The studies have shown how to estimate the perception of aliasing artefacts. The created method allows gathering a large quantity of aliasing perception data through the experimental procedure. Such a dataset can train the network, which would learn to distinguish noticeable distortions from invisible. However, in this work, the network has not achieved acceptable accuracy. Therefore, the temporal aspect of foveated rendering remains an open problem that requires future developments.

Author's contributions The foveated rendering method described in Section 2 was part of the project realised by a research team at the Max Planck Institute in Saarbruecken, Germany. I was an active participant in the meetings that took place regularly during the project, where we discussed progress made and ideas to pursue. I was responsible for programming the Unity and OpenGL implementations for standard and virtual reality displays and estimating their performance. I also carried out perceptual experiments for evaluating the quality of foveation. The project was published in [71].

The visual adaptation model from Section 2.3 was an individual work realised by me while overseen by my supervisor. I created a model, prepared its implementation and carried out an experiment to evaluate it. The findings have been published in [80]. I presented it at the International Conference on Multimedia Modeling 2019 in Thessaloniki, Greece.

The temporal aliasing artefacts detection method in Section 2.4 was proposed by me and worked on as a team project at the West Pomeranian University of Technology in Szczecin, Poland. I also prepared the neural network and the procedure for its training. I am solely responsible for creating a method and its evaluation.

Chapter 3

Learning perceived image statistics

This Chapter describes the process of creating and evaluating metamers through neural training and experimental procedures. The author's contributions in various project stages are listed at the end of the Chapter (see Section 3.4).

3.1 Objectives

As explained in the previous chapter, foveated rendering has been successfully used in the rendering process by differentiating between the fovea and periphery. Even though generating the resolution map using a predictor has provided valuable performance improvements, it does not rely on human perception entirely. The main problem it poses is the decrease of high frequencies amount in the image due to the resolution reduction. Even though such frequencies are not as critical in peripheral vision as in the foveal vision, it can be argued that they are still essential and provide meaningful information and are substantial in detail perception.

An example of high-frequency importance is shown in Figure 3.1. A structurally distorted image may look more realistic than the blurred one because texture changes in the far periphery may be less noticeable than the blur applied to it (see Section 1.5.1). The goal of this chapter is to determine the limitations and possibilities available through structural distortions.

3.1.1 Metameric images

Currently, metameric images are generated using two approaches. The first approach is to reduce the resolution of the rendered content in the periphery. Images received by the human visual system are identical to the full resolution, making

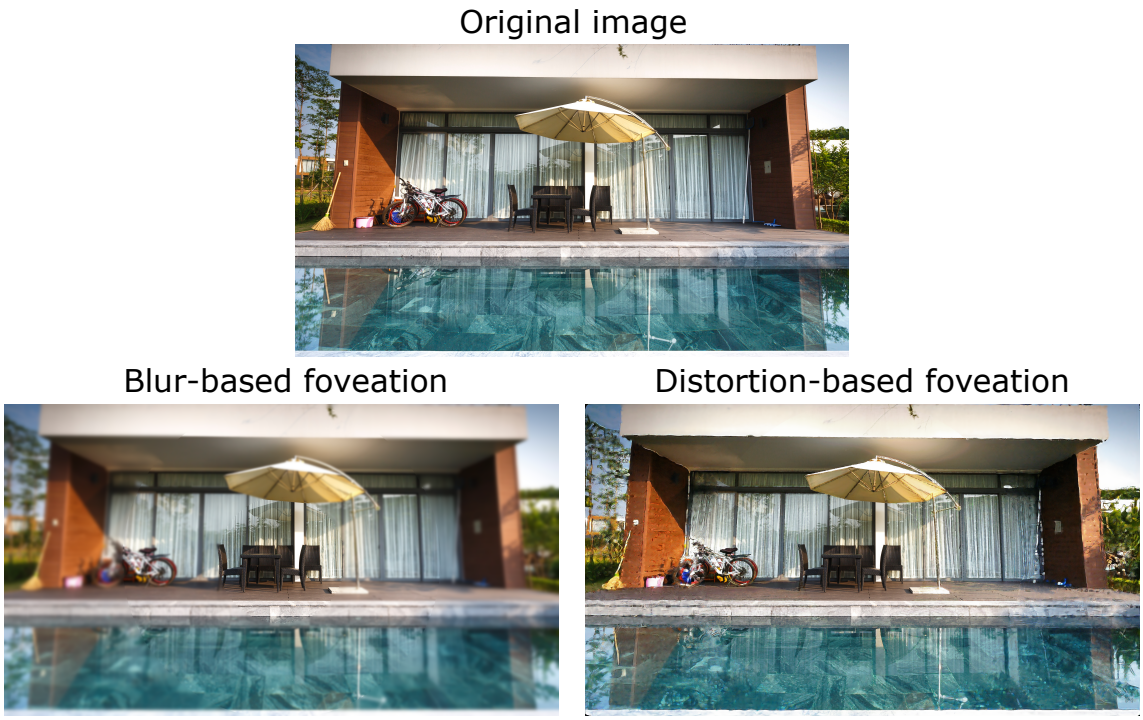


Figure 3.1: Comparison of standard foveation modelled using Gaussian blur, and foveation based on imperceptible distortions.

them the metamers. For more information on this particular technique, please refer to Chapter 2.

The second approach involves creating a neural network, which is supposed to learn image features and create metamers from a limited number of samples. An example of such a technique is DeepFovea [33]. It is based on the Generative Adversarial Network (GAN) [21] trained on ImageNet database [16]. The network received a video with only 9% of samples. It can recreate the video indistinguishable from the reference, including the temporal aspect passed using the recurring connections. Similar methods of training are introduced in neural supersampling methods [84, 34, 69]. Those techniques work by increasing the resolution of the original image through reconstructions of the missing details between the samples. With the recent rapid improvements in deep learning, such a reconstruction method became a popular feature in video games rendering, specifically with the NVIDIA DLSS [10] and AMD FidelityFX [3] super resolutions.

GAN training consists of two separate components: generative and discriminative. The generative network creates images from the given input (e.g. the number of samples or images at reduced resolution). The task of the discriminative network is to decide whether the input image is "fake" (i.e. created by a generative network) or "real". Simultaneous training of both networks is possible by introducing the min-max loss. This loss punishes the generator if the generated images were recog-

nised as "fake" and punishes the discriminator if they were considered "real". If set up correctly, this constant instability leads to continuous improvements in created image quality.

3.1.2 Contributions

Current deep learning super resolutions and foveation methods use full resolution images as the ground truth for the generator and as "real" image data for the discriminator. The results created through training have high perceptual quality and are successfully used for real-time rendering and improvements in performance compared to simple rendering in native resolution. Unfortunately, the usefulness of this technique is highly limited by the network's architecture. The DeepFovea model has 3.1 million parameters and requires four NVIDIA Tesla V100 GPUs to complete inference in 9 ms. Reducing the number of model parameters without decreasing the quality of created images would be an essential and beneficial part of deep learning rendering.

In currently used deep learning techniques, the images used as a reference are full resolution. The network attempts to create images with the highest possible quality, not necessarily only perceptually indifferent to the reference. This approach creates potentially unnecessary constraints. To combat this issue full resolution reference images could be replaced with metamers. This way, both the generator and discriminator would perceive the distortions in the reference as acceptable and would not punish them during the training. Such a scenario would relax the constraints on generated images, therefore decreasing the number of required network parameters.

In this Chapter, a way of creating such metamers is presented. Additionally, a robust comparison of training strategies is provided through experimental procedures to assess the quality of the proposed method.

3.2 System architecture

The system of creating metamers is based on the GAN training procedure [21]. The full scheme of the training system is shown in Figure 3.2.

Firstly, a set of input images at full resolution is prepared. Then, every image is distorted through the texture synthesis process (see Section 3.2.1). The levels of distortions are manipulated through the guiding samples, calibrated for human perception in the experiment (Section 3.2.1). Dataset prepared in such a way is the input to the discriminator part of the network, treated as the ground truth ("real").

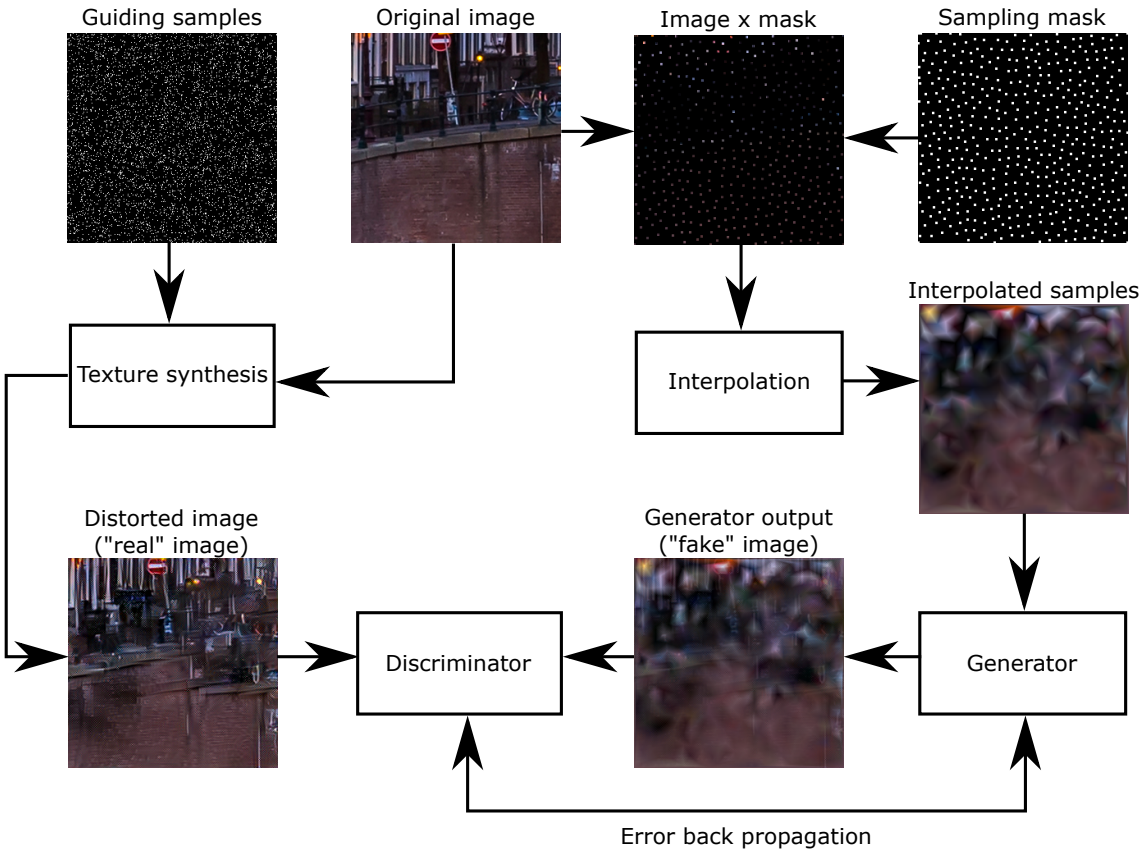


Figure 3.2: The architecture of the metamers generation.

Secondly, the input dataset for the generative part of the network is prepared. The input image is multiplied by a randomly generated sampling mask with a selected sampling rate. Generated samples are then linearly interpolated to fill in empty spots. Such images are used as input to the generator. The network's output is treated as a "fake" dataset by the discriminator. During the training process, the calculated error for the discriminator measures how well it recognises the difference between the "real" and "fake" images. The generator's error depends on the specific perceptual loss function and on its ability to convince the discriminator of the realness of the generated images. The whole training procedure is given in more detail in Section 3.2.2.

To account for different levels of acceptable distortions, two GAN trainings are performed: one for the near peripheral part of the vision, and one for the far peripheral part. This approach follows the idea shown in [23], where different parts of the image were rendered with different resolution settings. Therefore, the obtained training results are blended with the full resolution images so that the image's central (foveal) part has the highest acuity and the peripheral is significantly reduced. The boundary points have been selected according to human perception. A visualisation of this idea is presented in Figure 3.3.

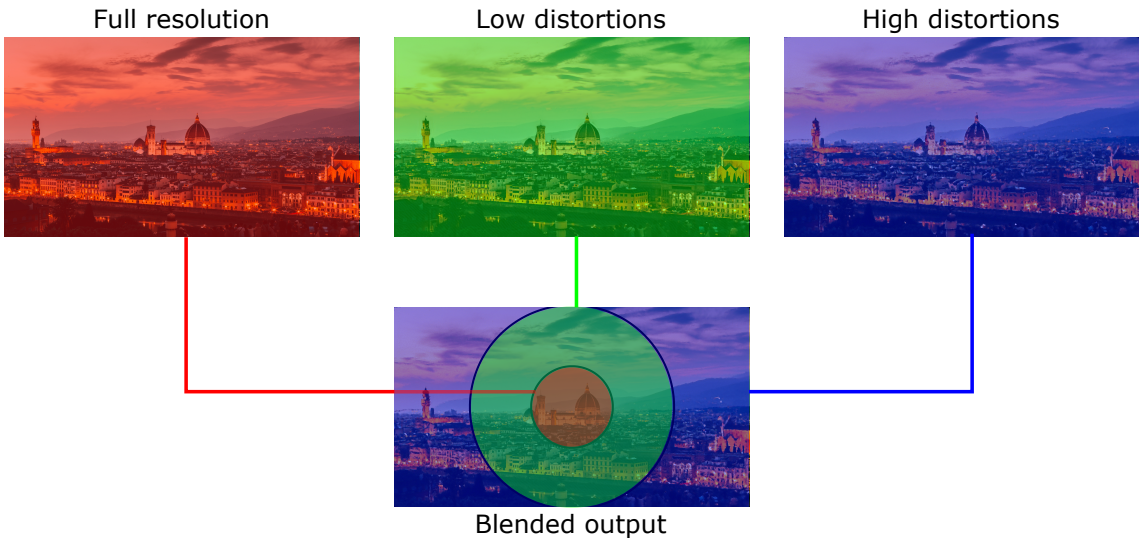


Figure 3.3: The generation of the full blended image by connecting original image with two networks' outputs.

3.2.1 Perceptually-driven image sampling

The main requirement for proper training using the proposed method is the metameric dataset. The images used as a reference would have to be distorted to a certain degree. The level of distortions would change depending on the eccentricity. At the end created dataset should be as distorted as possible without any of the distortions being detectable. To this end, an experiment was prepared in which the acceptable amount of not disturbing distortions was measured.

Creating metamers

A texture synthesis method proposed by Gatys et al. [20] was used to generate metamers. In the paper, the synthesis is computed through optimisation that iteratively minimises the difference between the input and reference. The optimisation is based on the VGG-19 network [65] which has been pre-trained on the ImageNet database, containing more than one million images. This network was originally used for the task of object classification. As a result, its pre-trained version is able to capture the most vital features of images needed for their recognition. Every layer corresponds to specific higher or lower level features. Using them for image synthesis constrains the output to have similar features as the reference but does not prevent it from creating significant displacements of pixels.

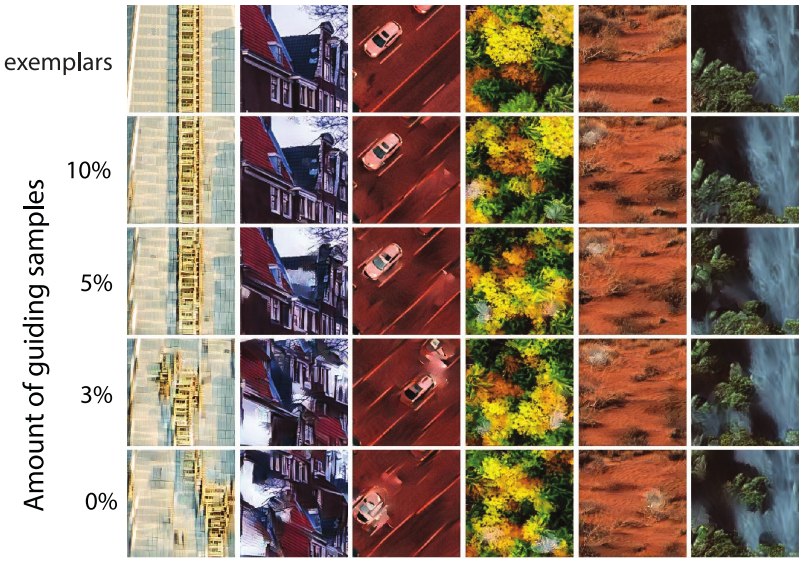


Figure 3.4: The results of texture synthesis performed for different number of guiding samples. The top row shows the reference image. The bottom row contains images without any guiding samples, i.e. with the original synthesis proposed by Gatys et al. [20] (reused image from own publication [68]).

For both the reference and input, the VGG features were computed on each of the 19 layers. A loss function for this procedure is given by the following formula:

$$\mathcal{L}(x, \hat{x}) = \sum_{l=0}^{L} w_l \sum_{i,j} \frac{1}{4N_l^2 M_l^2} (\hat{G}_{ij}^l - G_{ij}^l), \tag{3.1}$$

where x is the reference image, and $\hat{x}$ is the input. The loss is calculated as a sum of individual components across all L layers. Each layer l has a weight given by w_l. The calculations are based on all feature vectors extracted from the 3D-feature map for each layer. N_l is the total number of feature maps, and M_l is the number of neurons in the layer. For each feature vector defined by coordinates i and j, the Gram matrices $\hat{G}_{ij}^l$ and G_{ij}^l were computed. The difference between the matrices calculated for the reference and input indicates the magnitude of the loss function.

An additional component was added to the loss function to vary the level of distortions introduced by the synthesis. For every synthesised image, a random set of pixels was selected. During the optimisation, the values of those pixels were constrained for input $\hat{x}$ to have equal intensities as x. By choosing the percentage of all samples to be a constraint (further referred to as *guiding samples*), the levels of distortions in the final image could be manipulated. The results of this experiment are presented in Figure 3.4.

The above procedure allows the creation of images with structural displace-

Figure 3.5: The exemplar images used, along with their blurred versions. The amount of blur was adjusted by setting the Gaussian kernel filtering. The σ parameter was set according to the values given to the right of the first column of images (reused image from own publication [68]).

ments. An additional dataset was created that included images blurred using a Gaussian kernel to consider the loss of high frequencies due to the degradation of acuity in the peripheral region of the vision. The examples used in this work are presented in Figure 3.5.

Calibration experiment

Created dataset with various distortion levels was used for a calibration experiment. Its goal was to establish the maximal acceptable distortion depending on the distance to the gaze point.

Twenty images have been prepared for this trial. Half of them contained natural features, such as vegetation and landscapes, and half were artificial (buildings, cars, roads). The images were 256×256 patches gathered from the ImageNet database. A texture synthesis was performed for each patch with the number of guiding samples set to 0%, 3%, 5%, 7.5%, and 10%. During the visual inspection of the synthesised images, it was discovered that images generated with guiding samples set to a higher fraction than 10% lead to perceptually identical results with the reference. The blurred versions of the reference were also generated, with the σ parameter set to 0.25, 1.25, 2.25, 3.25, and 4.25. Overall, 200 test images have been prepared with 20 ground truths.

The participants were shown three images during the experiment: a reference in the centre, a distorted reference on either left or right side of the screen (selected

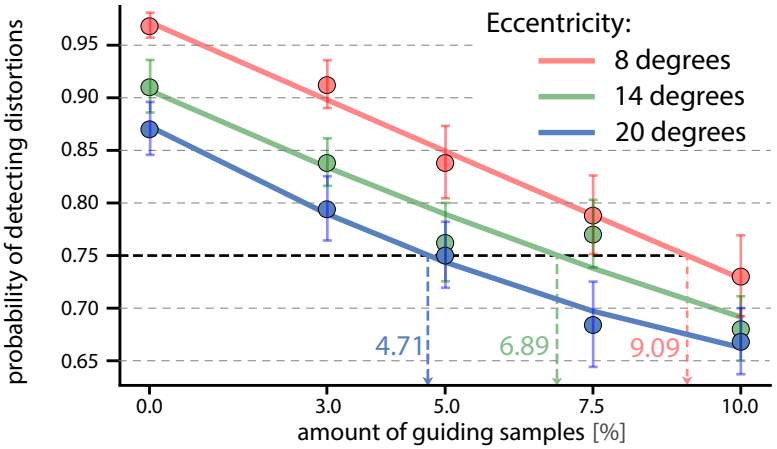


Figure 3.6: The detection probability of distortions. Dots represent the mean results across all images and participants. The whiskers indicate standard error of the means. Line connecting dots represent the interpolated function over the data points (reused image from own publication [68]).

randomly), and a reference on the opposite side of the distorted image. The user's task was to select which image (left or right) is identical to the reference. Images on the sides were shown at an equal distance from the centre. The user's gaze position was tracked using Tobii Eye Tracker. It was used to translate the whole stimulus such that the central image always remains at the gaze position. The eccentricity for the side images was randomly selected as one of the three: 8°, 14°, or 20°. Every set of stimuli was repeated five times in a random order, resulting in 3000 comparisons performed per participant. A total of five participants took part in this calibration experiment.

The results of the experiment are presented in Figure 3.6. They are divided into three groups, each associated with a single eccentricity. The probability of detection was calculated as the number of successfully detected reference images over all the trials. Then data points were interpolated to obtain the continuous detection probability function of distortions. As the last step, the fraction of the guiding samples translating to 75% detection rate (1 JND) was calculated. The calculated values are 4.71%, 6.89%, and 9.09% for 20°, 14°, and 8° eccentricity, accordingly.

3.2.2 GAN training

The data received from the experiments made it possible to establish the minimum number of guiding samples to make the images indistinguishable from the reference. The network was trained to generate such images by itself using this information.

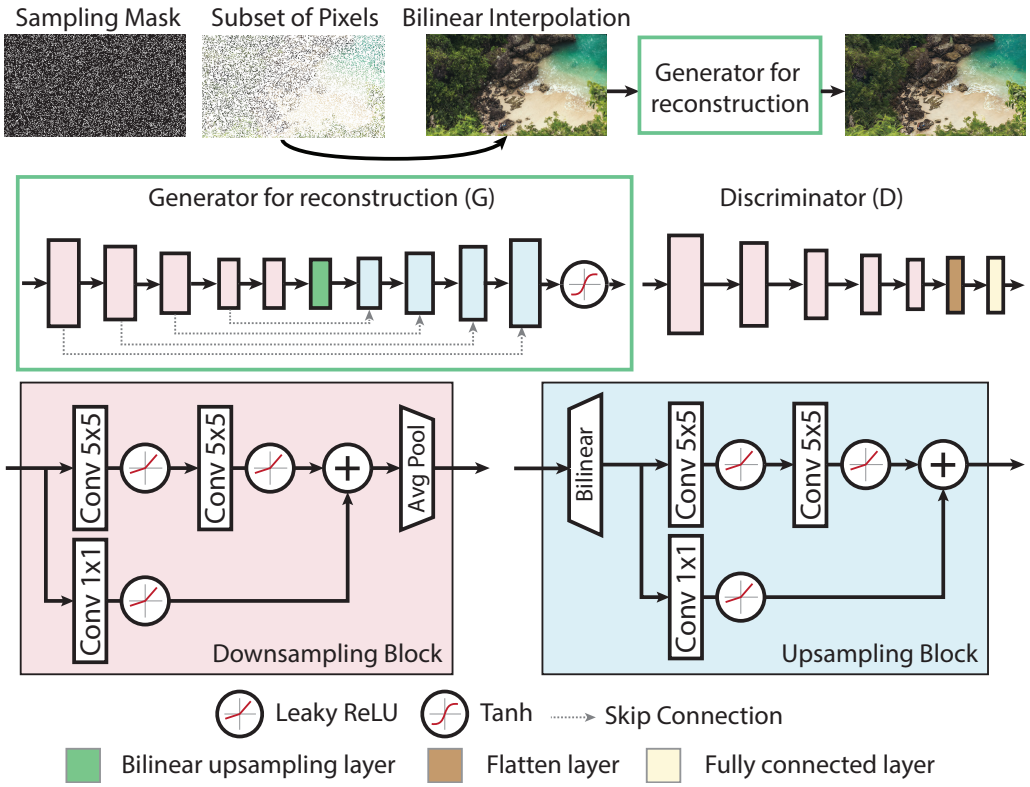


Figure 3.7: Scheme presenting the architecture of proposed network. Input image (billinearly interpolated subset of pixels) is passed directly to the first block of the generator (reused image from own publication [68]).

Training overview

The Wasserstein GAN [5] was used as a training algorithm. The scheme of it is presented in Figure 3.7. More details regarding the architecture are given in Sections 3.2.2 and 3.2.2 for the generator and discriminator, accordingly.

The model of the network is based on the U-Net architecture [63], which divides the networks into encoder and decoder parts. An example of such a design is employed by Kaplanyan et al. [33], on which the proposed network is based. Its purpose is to generate metamers from a limited number of samples given as input. As opposed to the standard GAN procedure, the ground truth data used for both generator and discriminator consists of distorted images that were obtained using the procedure from Section 3.2.1.

Generator

The generator network consists of 10 residual blocks, as shown in Figure 3.7 (*Generator for reconstruction (G)*). The topology of the first five blocks (i.e. the encoder blocks) is presented in the *Downsampling Block* section of the figure. The last operation in every block is average pooling, which downsizes the input by a factor of

two. The number of filters in blocks (from left to right) is 16, 32, 64, 128, and 128.

The other five blocks are decoder blocks (*Upsampling Block*). They perform similar operations as the encoder blocks but in reversed order. Additionally, the average pooling layer is replaced by the upscaling bilinear interpolation. The number of filters is the same as in the encoder blocks but reversed (from left to right: 128, 128, 64, 32, 16). The output of the final layer is then passed through the softmax activation, which creates a final output of the network.

Apart from the adversarial loss, a regulariser component is introduced. It rates the output based on its difference from the reference. It is referred to as *perceptual loss*. The network was trained in three ways, in which this difference was measured differently.

The perceptual loss was computed as the mean-squared difference between the output and ground truth in the first network. Such a loss function is known as the L^2 norm, further referred to as L2. For the second approach, the Learned Perceptual Image Patch Similarity index (LPIPS) was used [91]. LPIPS metric is computed by running the VGG network on a pair of images and computing the difference between outputs on all layers. Additionally, every layer has its linear weight set according to the experiments made by Zhang et al. The same weights were used to compute the perceptual difference between reference and network output.

The perceptual loss for the third approach is based on spatial frequencies. It is calculated by computing the Laplacian pyramid on the image and computing the mean squared error with the pyramid layers of the reference. Their importance in the output generation is adjusted by weighing each layer differently, giving more freedom in hallucinating details for specific frequencies. This approach is based on the process of enforcing perceptual consistency proposed by Hepburn et al. [24].

Discriminator

The discriminator's architecture is similar to the decoder of the generator, as presented in Figure 3.7 (*Discriminator (D)*). The first five blocks are down-sampling the data. They are followed by the flattening and dense layer, producing a single scalar per input. The PatchGAN discriminator [28] was used by splitting the input into 64×64 patches to focus more on the local features of the image. The final prediction for the image is calculated as a mean of all the patches output.

For evaluating the predictions, standard Wasserstein GAN loss is employed. It is defined as:

$$\mathcal{L}_{adv} = D(x) - D(G(z)), \tag{3.2}$$

L2	$\mathcal{L}_G^{L2} = w_{L2} \cdot \mathcal{L}_{L2} + w_{adv} \cdot \mathcal{L}_{adv}$
L2 ours	$\mathcal{L}_{G^*}^{L2} = w_{L2} \cdot \mathcal{L}_{L2} + w_{adv} \cdot \mathcal{L}_{adv}^*$
LPIPS	$\mathcal{L}_G^{LPIPS} = w_{LPIPS} \cdot \mathcal{L}_{LPIPS} + w_{adv} \cdot \mathcal{L}_{adv}$
LPIPS ours	$\mathcal{L}_{G^*}^{LPIPS} = w_{LPIPS} \cdot \mathcal{L}_{LPIPS} + w_{adv} \cdot \mathcal{L}_{adv}^*$
Laplacian	$\mathcal{L}_G^{Lapl} = w_{Lapl} \cdot \mathcal{L}_{Lapl} + w_{adv} \cdot \mathcal{L}_{adv}$
Laplacian ours	$\mathcal{L}_{G^*}^{Lapl} = w_{Lapl} \cdot \mathcal{L}_{Lapl} + w_{adv} \cdot \mathcal{L}_{adv}^*$

Table 3.1: The loss functions used for generator training. The first column indicates the type of training performed - L2, LPIPS, and Laplacian are training with mean-squared error, VGG outputs, and Laplacian pyramid, accordingly, as explained in Section 3.2.2. Annotation *ours* is related to training with distorted images as reference. Absence of it means training on standard full-resolution images.

where $G(z)$ is the output of the generator G run on the input z, x are the reference images, and $D(..)$ is the discriminator prediction.

Training

A proposed training procedure makes the networks specialised for specific eccentricities. The viewing area is divided into three regions: fovea at 0°–8°, near periphery at 8°–14°, and far periphery beyond 14° [23]. Each of the regions has an assigned amount of samples that are required for full resolution perception. The results from the experiments in Section 3.2.1 were used conservatively. The foveal region is kept in full resolution (100% samples) without introducing any distortions. 9.09% of samples were used for the near peripheral region and 6.89% for the far peripheral.

The images used for training have been taken from the ImageNet database. 50k images were selected randomly, 50 per each of 1000 available classes. 256×256 patch was cropped at a random location from each image. For every training procedure, unique masks were created with samples corresponding to the selected sampling rate: 0.7% for the far periphery and 12% for the near. All samples in masks have then been rearranged using a void-and-cluster algorithm in order to spread them evenly across the patch [72].

As mentioned in Section 3.2.2, networks were trained with various perceptual loss functions. Additionally, in order to compare proposed method to standard approach, all networks were trained with and without distortions presented in the reference images. All used generator losses are listed in Table 3.1. Weights used for the training are: $w_{L2} = 2000$, $w_{LPIPS} = 100$, $w_{Lapl} = 100$, $w_{adv} = 1$. As an optimizer, an ADAM [37] was used with a learning rate 2×10^{-5}. The training proceeded until the loss value for both generator and discriminator stopped decreasing, which took between 20 and 30 epochs.

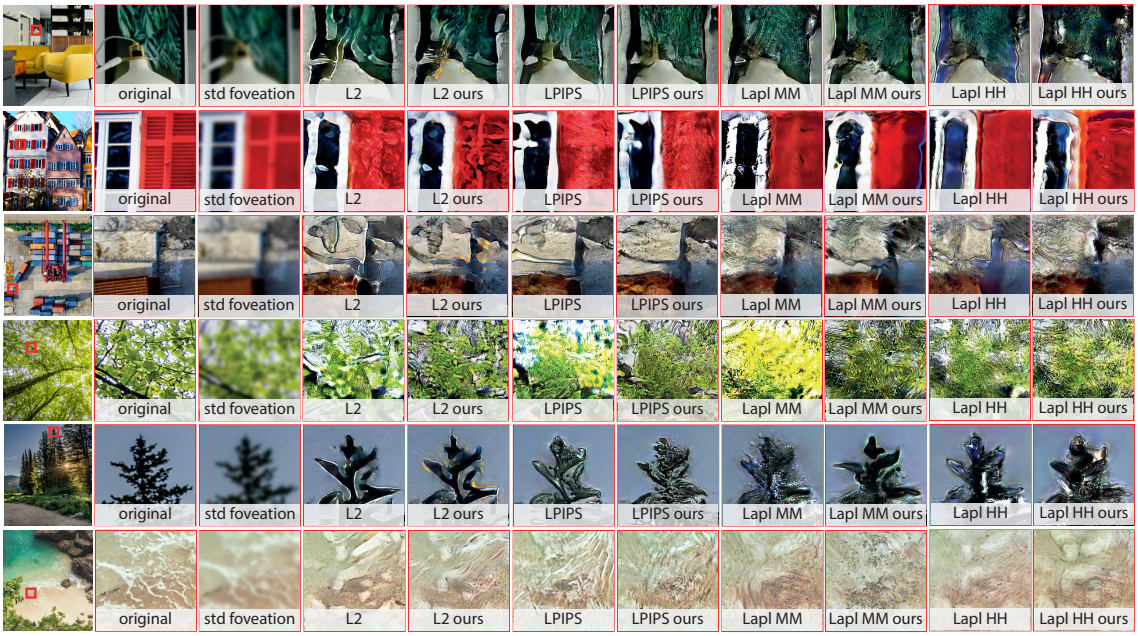


Figure 3.8: The outputs gathered from all training procedures. The first column shows full resolution images not used for training. All other columns show zoomed-in parts of the patch for different scenarios. *std foveation* shows images after applying Gaussian blur corresponding to the human visual system capacity at a given eccentricity. The two letters following the *Lapl...* images indicate the distribution of weights across all Laplacian pyramid layers. *MM* means the highest weights are set to medium frequencies (fourth pyramid layer). For *HH* the weights are the highest at the highest frequencies (top pyramid layer). The first letter corresponds to the weights set for the far periphery, the second for the near periphery (reused image from own publication [68]).

3.3 Evaluation and results

The sample results from all the performed training procedures are presented in Figure 3.8. All the outputs present some level of distortion. They seem to appear due to hallucination, not reconstruction – the exact details are not recreated; instead, they form structures not present in the original patches. Additionally, unlike the standard foveation using Gaussian blur, they contain a wide range of frequencies.

Training procedures that use L2 as their loss term produce images with more uniform areas and distinct edges. Laplacian pyramid training with the highest weights set to medium frequencies (*Lapl MM*) shows the highest discrepancies compared to the original in terms of high frequencies placement – they appear in seemingly random locations. The high frequencies are not as constrained as lower frequencies; therefore, the network has more freedom in creating them. On the other hand, training with the highest weights set to high frequencies (*Lapl HH*) does not create an expected better quality of high frequencies. It is arguably caused by the

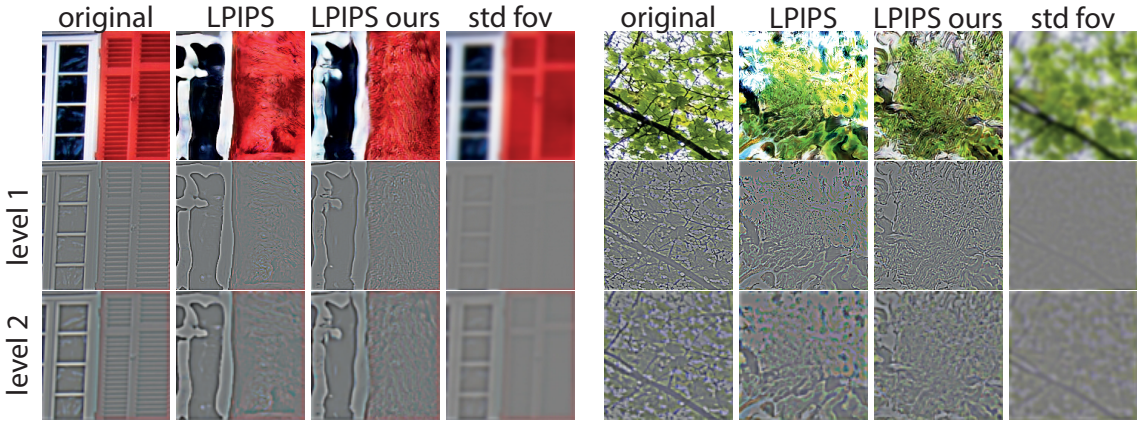


Figure 3.9: The comparison of high frequencies amount between the original patch, training results, and blurred using Gaussian kernel. Two bottom rows (level 1, level 2) represent the first two layers of their Laplacian pyramids (reused image from own publication [68]).

difficulties in recreating high frequencies. They appear in the highest amount in very distinct locations, making it hard for the network to evaluate. It hampers any possibilities of decreasing the loss significantly. Therefore, the only significant improvements are observed for lower frequencies. LPIPS-based images have the least amount of distortions in comparison to other methods. They also have a similar spatial frequency distribution to the original patches.

The frequency distribution is different when comparing the results with standard images given as input to the results from the distorted input set. This phenomenon is evident in the LPIPS, where substituting the input set increases the number of high frequencies. It is more clearly visible after separating the patch into Laplacian pyramid layers, as shown in Figure 3.9. Patch with blur applied to it contains, as expected, the fewest high spatial frequencies. The difference between LPIPS and LPIPS Ours is the most pronounced in the first pyramid layer.

Including in the training images with perceptually invisible distortions could produce results that look more natural than using a standard dataset. Two tests were performed to verify it. A metric comparing images and measuring their difference to the reference was created in the first one. In the second test, a user study was performed where the participants could directly assess the quality of all images.

3.3.1 Perceptual quality metric

Due to the lack of universal solutions, new quality metrics were created based on existing methods, adjusted for this study. During the calibration experiment (Section 3.2.1), robust data about the distortions detection rate was gathered. The exact fraction of people who detected each patch's distortions is known. A set of

functions was created to compute the predicted detection value for distorted images to generalise the findings on any patch.

The quality metrics are based on existing metrics: L2, SSIM [74], MS-SSIM [76], and LPIPS [92]. An additional metric is based on the custom LPIPS, where each weight of the VGG-19 layers was recalibrated. It is further referred to as Calibrated VGG. Every *reference-distorted patch* pair was converted into a scalar value using all methods. Then it was fitted to the psychometric logistic function [62]:

$$y(t) = a + (k - a)/(c + q \cdot e^{-b \cdot t})^{\frac{1}{v}}, \tag{3.3}$$

where a, b, c, k, q, and v are the free parameters that require calibration.

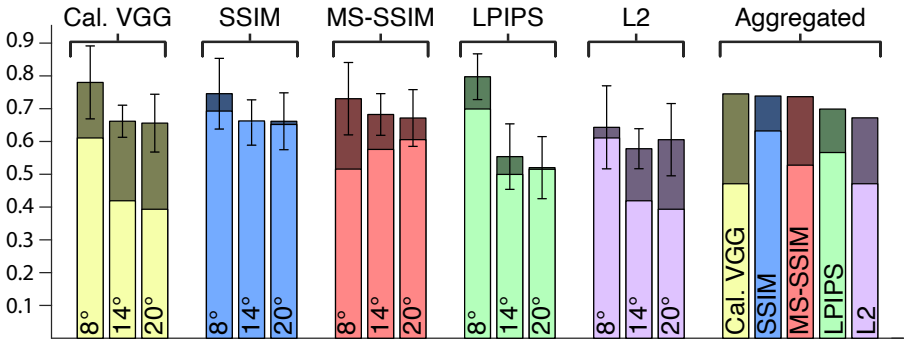


Figure 3.10: Pearson's correlation coefficient for all tested methods and eccentricities. Bright bars indicate results for uncalibrated functions in the sigmoid logistic function. Darker bars represent calibrated logistic functions based on the minimisation algorithm. The last section (Aggregated) shows results where the correlation was calculated for images from all eccentricities jointly. The whiskers show the standard deviation (reused image from own publication [68]).

To get the best values for all the parameters of the logistic function and the weights of VGG layers in the Calibrated VGG, the minimization was performed. The problem can be formulated as:

$$\min_{p} \sum_{(x,\hat{x}) \in S_r} \|y(M(x,\hat{x}), p) - P(x,\hat{x})\|^2, \tag{3.4}$$

where p is the set of parameter values, S_r is the dataset containing all pairs of patches for a given eccentricity r, $\hat{x}$ is the distorted version of patch x, M is one of the metrics (L^2, SSIM, MS-SSIM, LPIPS, or Calibrated VGG), y is the psychometric function, and P is the probability of detecting the difference between patches. This function was minimised using *trust-region-reflective* [7] and *Levenberg-Marquardt* [45] optimizations. The experiment results are presented in Figure 3.10 as a Pearson's correlation coefficient between real and predicted probability of detection. The cal-

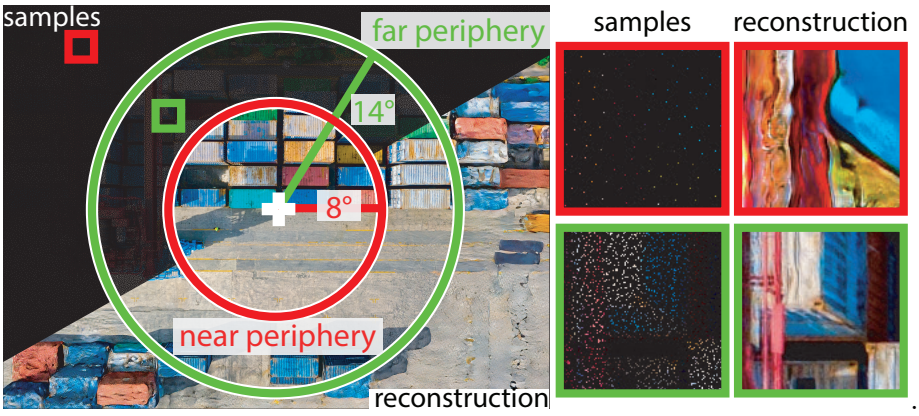


Figure 3.11: An exemplar image generated by combining the outputs of networks and the reference (reused image from own publication [68]).

ibration was done using five-fold cross-validation to consider potential overfitting problems. 80% of the available data was used for calibration. The reported results come from the remaining 20%, gathered across all five folds.

The results show that the predictions for higher eccentricities are lower than for stimuli placed at the near periphery. This difference is evident for the LPIPS case, where the correlation is over 0.2 lower for 14° than for the 8° case. This phenomenon is caused by the fact that all tested metrics have been designed initially for direct observation only. Any limitation posed by the human visual system's peripheral capabilities is not considered. However, this problem is nonexistent for the calibrated VGG. All layer weights have been tailored for a specific eccentricity, which proved successful, especially when comparing calibrated and uncalibrated setups. Overall, when comparing the aggregated results, the calibrated VGG performs best.

The metric was adjusted to calculate the predicted detection rate for any image as the next step. The metrics were combined for all eccentricities by linearly interpolating predictions for 8° and 20°. The prediction can then be performed in the following procedure: firstly, an image is divided into 256×256 patches. Each patch is assigned a value equal to the eccentricity from the viewing position (e.g. centre of the image). Then, the prediction is calculated for each patch separately. The final prediction is equal to the mean of all patches.

Ten publicly available images were acquired. They were sub-sampled and used as an input for all trained networks. Then the full resolution images were connected with near and far periphery outputs by blending them. The centre (up to 8° eccentricity) was kept at full resolution, area 8° – 14° contained the output of the near periphery network, and the remaining region outside the 14° was generated using the far periphery network. An example of a final image generated this way is presented in Figure 3.11.

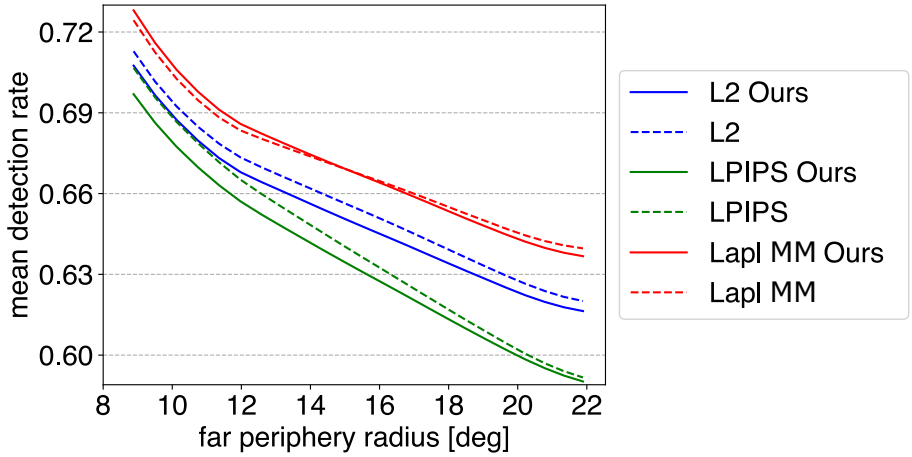


Figure 3.12: The predicted detection of differences between all reference and generated images for the trained networks (reused image from own publication [68]).

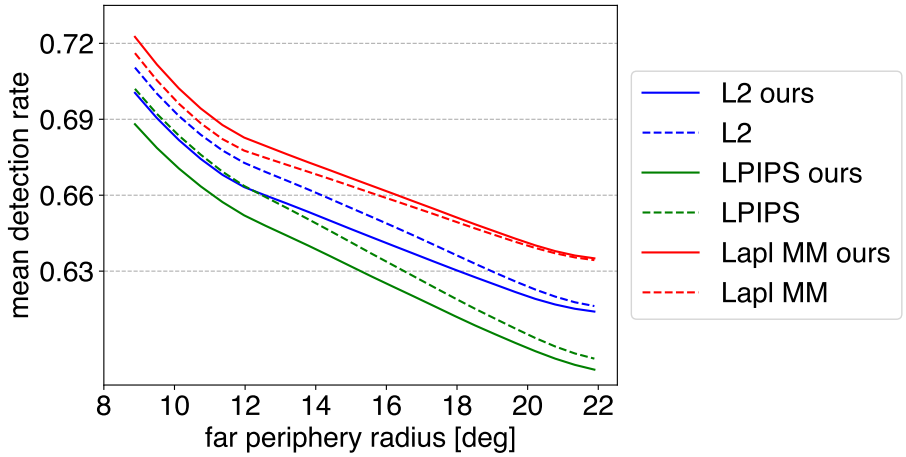


Figure 3.13: The predicted detection of differences between reference and generated images with natural features for the trained networks (reused image from own publication [68]).

The reconstruction quality was tested by running the metric on all of the generated images. Additionally, other eccentricities ranging from 9° to 22° were tested to check the impact of the choice of the far periphery location. The results of this test are presented in Figure 3.12. Out of all tested methods, LPIPS ours reached the lowest predicted detection rate, i.e. the images created using this training would look the closest to the reference. The difference between methods is the most apparent if the starting point of the peripheral image is set to a high eccentricity. The LPIPS ours consistently outperforms standard LPIPS. The test dataset was separated into images with artificial and natural features to analyse the detection rates further. The results for the natural images are shown in Figure 3.13. In comparison with Figure 3.12, the overall predictions are lower for all tested methods. Furthermore, the difference between standard and distorted dataset predicted detection rates is

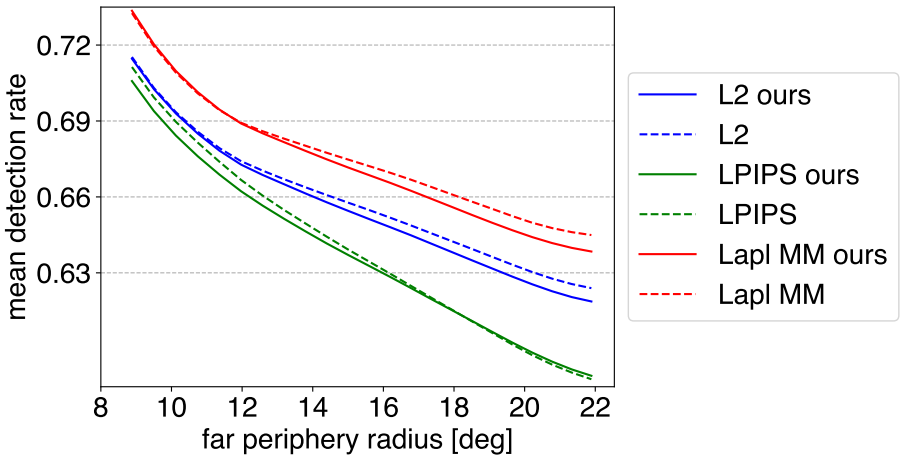


Figure 3.14: The predicted detection of differences between reference and generated images with artificial features for the trained networks (reused image from own publication [68]).

larger, which shows a more considerable advantage of selecting a distorted dataset. On the other hand, this trend is reversed for the images with artificial objects. As shown in Figure 3.14, the predicted detection rates are higher. Additionally, the difference between distorted and standard training inputs is minuscule. Such a discrepancy between images with artificial objects and natural scenes is thought to be caused by the differences between structural features. Artificial objects often contain large structures. Because the patches used for training were limited to 256×256 size, such structures might be only partially represented. The network was trained on patches specifically, not accounting for any structures spanning more expansive areas. Because of that, the network could not improve its predictions for larger areas. In contrast, natural images contain a lot of chaotic regions without any clear structures. Therefore, the patch size selection had a much lower impact on them.

3.3.2 Subjective experiments

Apart from running the metric on the selection of images, perceptual experiments were performed, where the participants could directly compare the quality of reconstruction. Two images were presented on display for a single trial: reference on the left and reconstruction on the right. By pressing a key, the participants could switch the right image between reconstructions created with the proposed method and with standard training. Their task was to select an image that looked the closest to the reference. The following combinations of stimuli were tested: LPIPS vs LPIPS ours, LPIPS vs L2, LPIPS vs L2 ours, LPIPS ours vs L2, LPIPS ours vs L2 ours, and L2 vs L2 ours. The procedure was completed for ten images, which requires 60 trials per observer. The trials were presented in random order. A total of 15 participants

Table 3.2: The fractions of trials in which the proposed method was preferred in comparison to standard. Every value was calculated for the specific type of used generator and set of images. Values in parenthesis signify the p-value of the selected samples. Values shown in bold are statistically significant.

Techniques \ Image set	Natural	Artificial	Full
L2 Ours vs L2	**0.75** ($p < 0.01$)	**0.37** ($p = 0.04$)	**0.56** ($p < 0.01$)
LPIPS Ours vs LPIPS	**0.75** ($p < 0.01$)	0.43 ($p > 0.10$)	0.59 ($p > 0.10$)
All Ours vs All	**0.75** ($p < 0.01$)	**0.40** ($p < 0.02$)	**0.57** ($p = 0.01$)

Table 3.3: The fractions of trials in which the method listed in the first column was preferred over the others. Values in parenthesis signify the p-value of the selected samples. The values shown in bold are statistically significant.

Technique \ Image set	Natural	Man-made	Full
LPIPS Ours	**0.68** ($p < 0.01$)	**0.58** ($p = 0.03$)	**0.64** ($p < 0.01$)
LPIPS	**0.37** ($p < 0.01$)	**0.62** ($p < 0.01$)	0.48 ($p > 0.10$)
L2 Ours	**0.62** ($p < 0.01$)	**0.35** ($p < 0.01$)	0.48 ($p > 0.10$)
L2	**0.33** ($p < 0.01$)	0.45 ($p > 0.10$)	0.40 ($p > 0.10$)

were tested, naive to the purpose of the experiment. The experiment results are shown in Table 3.2. Overall, the proposed methods were preferred in comparison to the standard. However, the individual results are a bit more convoluted. The proposed method is strongly preferred in the case of images with natural features. This result follows the findings involving the quality metric (Section 3.3.1). As the reference images contain many chaotic patterns, the network is not punished for creating them, while the standard network is. Such images are simpler to generate because learning to create a metamer is more straightforward than creating a full resolution image.

On the other hand, the presence of artificial objects shifts the preference to standard methods. As mentioned before, it is expected to be caused by the lack of larger structures in the training database. The results showing the preference of individual methods are provided in Table 3.3. They show that images with natural features look better if generated using the proposed method. This statement holds for both LPIPS and L2 training procedures. LPIPS-based methods provide the best results for images with artificial objects, with a slight preference for the standard LPIPS method. Overall, the LPIPS method with a distorted dataset is the most preferred out of all the others.

3.4 Chapter summary

During this study, a method of injecting the perception directly into the discriminator of the GAN system was created. By distorting the input data, the perceptual capabilities of the human visual system were simulated at different eccentricities. It relaxed the constraints imposed on the network and allowed it to create images more freely.

In the end, the images created through the training using the proposed method turned out to be of higher perceptual quality than generated through the standard method, as proven in the study. These findings may be used in the future to simplify the network and replace the input dataset to generate high-quality images faster.

However, due to the large structures present in artificial objects, the network failed to create better quality images containing artificial bodies. Increasing the size of input patches and expanding the dataset should provide better results.

Author's contributions A research team completed the project described in this chapter at the Università della Svizzera italiana in Lugano, Switzerland, cooperating with the Max Planck Institute and the West Pomeranian University of Technology. I was active in designing the method and took part in regular meetings during the project. I was a significant contributor to creating a network, training it and preparing experimental procedures. I have also programmed the perceptual quality metrics and compared their effectiveness.

Chapter 4

Improving multi-focal rendering performance

This Chapter describes the process of rendering on the multi-focal displays. A new hybrid approach is explained, which improves the performance significantly without negatively affecting the visual quality in comparison to the previously developed solutions. The author's contributions in various project stages are listed at the end of the Chapter (see Section 4.6).

4.1 Background

Vergence-accommodation conflict. A fundamental problem that cannot be solved using classic VR goggles is the *vergence-accommodation conflict* [25]. Although VR goggles provide a high level of realism, they cannot simulate eye accommodation. A human lens can change its shape by modifying the focal length value. In the case of goggles, the lens has a predetermined focal length, so the observed three-dimensional scene is always at the same range from the eyes. By converting the focal length formula, the distance of the virtual image from the lens can be calculated as: $v = 1/(1/d - 1/f)$, where d is the distance from the display to the lens, and f is the focal length. Regardless of the rendered content, the eyes are focused on the virtual image, which makes objects out of focus blurry to a certain degree. For visualisation of this phenomenon, see Figure 4.1. This issue does not affect far objects to the same extent as those closer to the observer due to the resolution limitations – all diverged rays would come from the same pixel.

The discrepancy between accommodation and binocular disparity does not allow for an entirely correct simulation of 3D vision. Two solutions have been proposed to solve this issue. The first idea involves using varifocal lenses that can change their shape and, consequently, focal length. This approach requires continu-

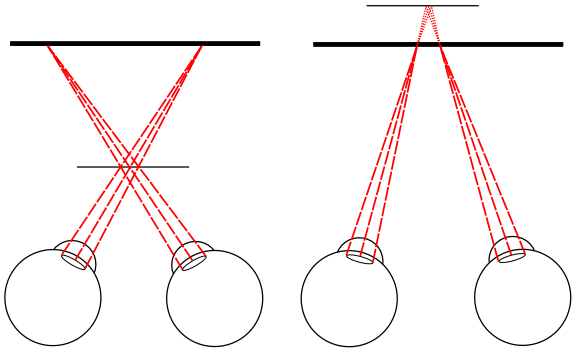
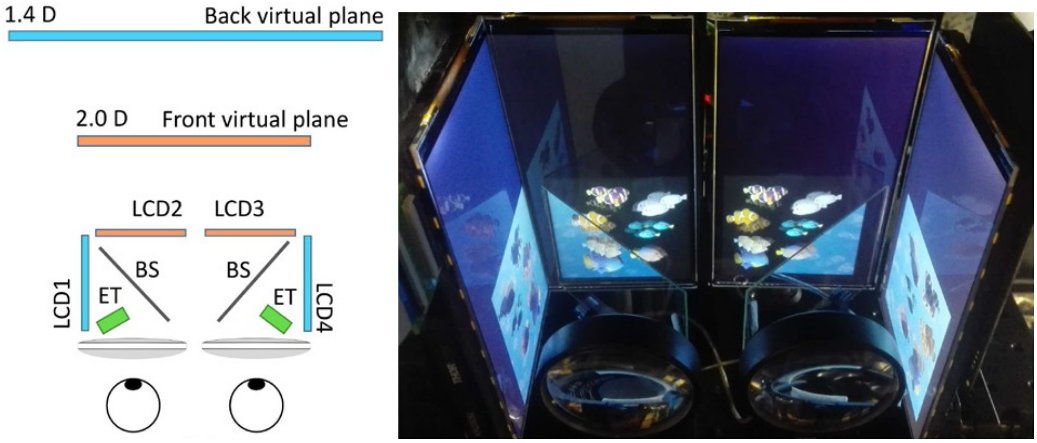


Figure 4.1: The vergence-accommodation conflict present in the VR setting. Thick black lines represent the virtual plane, on which the eyes are focused. Visible objects (thin black lines) would appear blurry to the user due to the fact, that light rays do not cross at their surfaces.

ous eye and gaze position data to establish the required focus distance. The second option is to use a multi-focal display. With multiple viewing planes presented to the observer, one could focus on any object in the region between them. An example of such a display is the system used in this study, shown in Figure 4.2.



(a) (b)

Figure 4.2: (a) The scheme of multi-plane display and (b) the photo of a setup used for the multi-focal experiments. The beam splitter reflects and refracts the light coming from the LCD screens. Slight shift in the placement of the displays and sizes of the generated images results in projection of images into two virtual planes (reused image from own publication [87]).

Multi-focal display. The image appears on four screens, two per eye, contrary to an ordinary display. Between each pair of screens, there is a beam splitter that reflects half of the light and refracts the other half. Since the screens on the sides are slightly further away from the lenses than the screens in front, the images they generate appear farther away. The dioptric distances of a near and far plane are 2.0D and 1.4D, accordingly (corresponding to 0.50m and 0.71m).

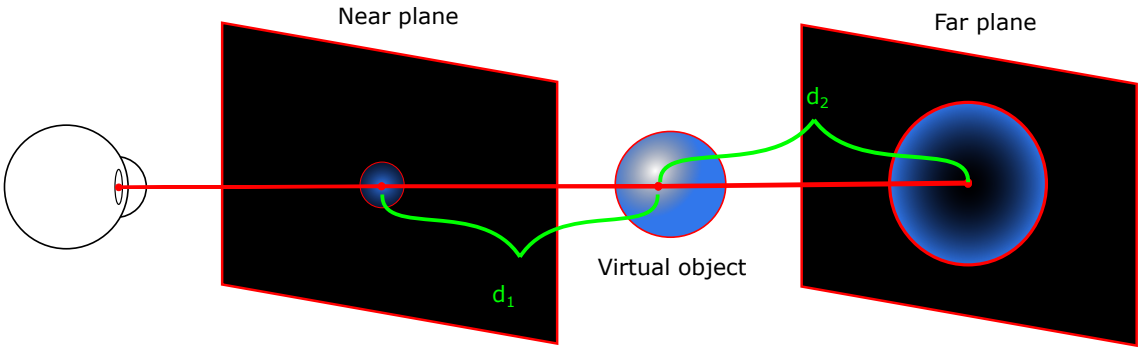
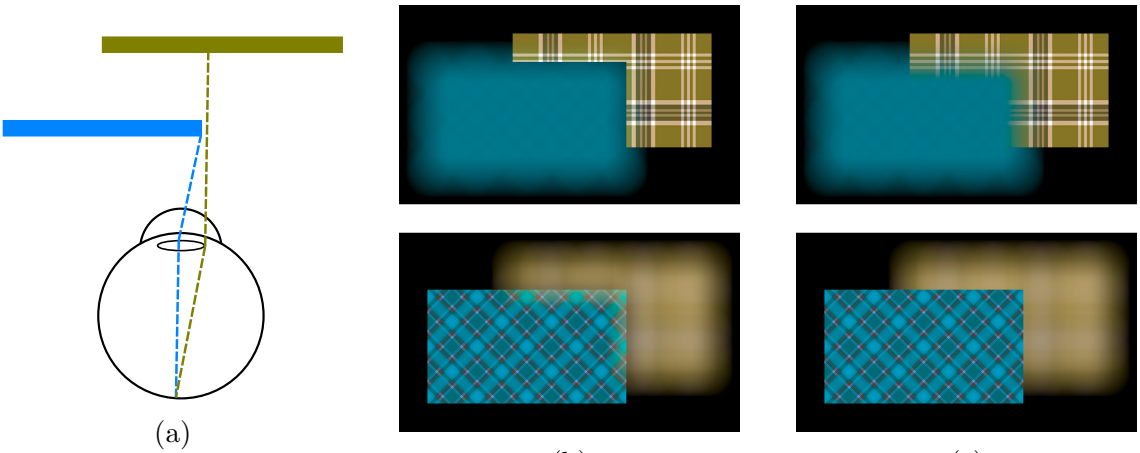


Figure 4.3: An example of decomposition used to render 3D objects. The variables d_1 and d_2 measure the distance between the position of the object point in virtual space and the corresponding pixel position in near and far plane, respectively.

An example of the rendering process for a multi-plane display is presented in Figure 4.3. The rendered image is split into two displays so that the objects close to the observer ($d < 0.50m$) are on the near plane, and the more distant objects ($d > 0.71m$) are on the far plane. Objects between two planes are rendered simultaneously on both screens, with an intensity ratio depending on the distance between them. The intensity sum is the same for any given degree of separation. Because of that, the colours of the objects do not deteriorate. Additionally, the displays must be perfectly aligned with the eye so that the object's fragments overlap for both planes. The exact position of the user's head is required to achieve this effect. A calibration procedure has to be performed for every user to measure it, during which the corners of near and far displays have to be aligned.

Rendering for multi-focal display. The most crucial step in the rendering process for multi-focal displays is arguably decomposing the rendered image into two layers. The simplest way of distributing intensities between planes is to perform a linear blending (LB) based on the depth [2, 25, 50, 64]. The intensities of every fragment are multiplied by the weight equal to $d_2/(d_1 + d_2)$ for near plane, and $d_1/(d_1 + d_2)$ for far plane (see Figure 4.3).

Problem. LB is fast and accurate enough in most situations; however, its design limits it. The observer is modelled as a pinhole camera, with minimal aperture and without any blur introduced by focus changing. It introduces problems, especially on the object boundaries, as shown in Figure 4.4, (a). Suppose the user is looking straight forward and is focused on the blue object. In the real world, the rays from both green and blue objects reach the retina by passing through all points within the eye lens. In LB the depth is calculated for the light rays coming through the centre of the lens only. Because of that, the ray marked in the figure as green would



(a) (b) (c)

Figure 4.4: (a) An example of mixing light rays from different objects into a single photoreceptor. (b) Occlusion artifacts created while using linear interpolation blending, compared to (c) the real-world scenario. Top images visualize focusing on the back green texture with vertical stripes, bottom on the front blue with diagonal patterns. Images are based on Figure 3 from the paper by Narain et al. [54].

not be considered in the rendering process.

In general, whenever the depth changes drastically in neighbouring pixels because of occlusions, LB introduces visual artefacts [54], as shown in Figure 4.4, (b) and (c). In the case of far plane focusing, the pixels in the occlusion region belonging to the foreground seem to appear in focus. On the other hand, focusing on the near plane introduces halo effects from the pixels near the occlusion. Objects displayed further away are partially visible in the area of focus.

Potential solutions. Because of its limitation, better decomposition methods than LB would be desirable. The first approach is the light field synthesis (LFS) [26, 44]. The bases of this technique are the 4D light fields [46, 22]: vector functions describing the radiance of the light rays coming from every point on the screen in every direction. To approximate the light fields of the rendered scene, it has to be generated from multiple points of view, shifted in vertical and horizontal directions. Those extra calculations drastically increase the performance cost of the method. On the other hand, it solves the issue of occlusion artefacts present in the LB [89].

Alternative approach is based on the retinal optimization [53, 54]. It involves rendering a series of images, each presenting a scene while focusing on objects at a specific set of distances. All those images create a so-called *focal stack*. Out of those images, one can acquire the final rendering through optimisation techniques. They can then be presented on the visual planes to match all accommodation distances best. The limitation of this method is similar to the LFS: it increases the overall performance cost. It requires generating images and performing optimisation to

calculate the plane decomposition.

Every approach mentioned above has its advantages and disadvantages. The ideal solution would combine all three methods to create a decomposition closest to the ground truth and with acceptable performance. Unfortunately, there have not been any studies comparing their properties and quality levels. As the main contribution of this work, a new hybrid decomposition method is proposed. It was created by evaluating the existing techniques and selecting the one which enables the highest perceivable quality.

4.2 Improved image rendering for multi-focal display

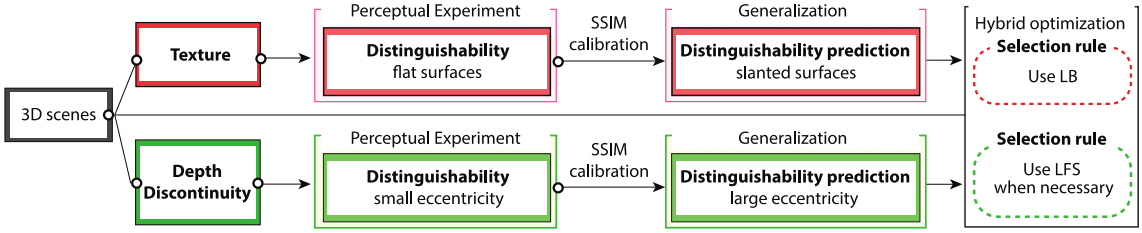


Figure 4.5: The overview of the proposed method. The texture and depth discontinuity of the rendered scenes are analysed through perceptual experiments. Then, using SSIM calibration, the model is generalised for all types of stimuli. The hybrid optimisation is computed by selecting the appropriate method following the calibrated metric. In the end, the output is compared to the ground truth (reused image from own publication [87]).

Due to the requirement of rendering a scene from multiple points of view and applying an optimisation, LFS is much more expensive than LB. Therefore, if the choice has to be made between those two methods for decomposition, the following rule should be applied:

- If LB and LFS produce perceptually identical results, the LB method should be selected.
- If LB and LFS produce different results, the selection should be based on their difference compared to ground truth.

The performance of LFS can be increased by replacing some of its parts with LB using this rule. Such a hybrid method would come with no loss of quality.

The overview of this method is shown in Figure 4.5. This idea is based on the measurements of the quality of generated images from the perceptual point of view. It can be done by performing a perceptual experiment.

The reproduction quality of LB and LFS was compared in various situations by analysing two main issues in multi-focal display rendering: the inaccuracies in texture mapping and depth discontinuities. The users observed two stimuli generated with different methods. Using calibrated SSIM [75] metric, a method of predicting the detection rate for other stimuli with similar features was created. Such an evaluation has proven to be viable in the previous studies [38, 82].

4.3 Experimental evaluation of decomposition methods

The hybrid decomposition relies on two methods: LFS and LB. In terms of performance, LFS is significantly more expansive than LB. It does, however, come with higher quality which was why it was decided to be the default choice.

LB decomposition shows issues with images containing occlusions [54]. However, there are some areas in which LB performs on a similar level of quality to LFS. Understanding the rules governing this phenomenon would allow assigning methods to specific image areas. To this end, an experimental procedure was prepared.

The experiment was divided into two parts. The first part analysed the reproduction of textures and the effect of method selection on quality. The second part looked into depth discontinuities to understand how both methods deal with them.

4.3.1 Texture distinguishing experiments

Perceptual experiment

Various spatial frequencies ranging from 6 to 15 cycles per degree were analysed to test the texture quality in LB decomposition. For each selected frequency, the participant observed two pairs of sinusoidal patterns: the first pair consisted of two stimuli generated using LFS, and the second pair had a stimulus generated with LFS and a stimulus generated with LB. LB stimuli were generated by rendering a single stimulus and decomposing it using a depth map. LFS generated additional eight views around the centre. The task was to select the pair which contained two different patterns. In the first experimental procedure, the stimuli were shown on the flat surface only, perpendicular to the observer. They were placed randomly on the top and bottom rows of the display, as shown in Figure 4.6, (a). A total of five participants with normal or corrected-to-normal vision took part in the test. All of them were naive to the purpose of the experiment.

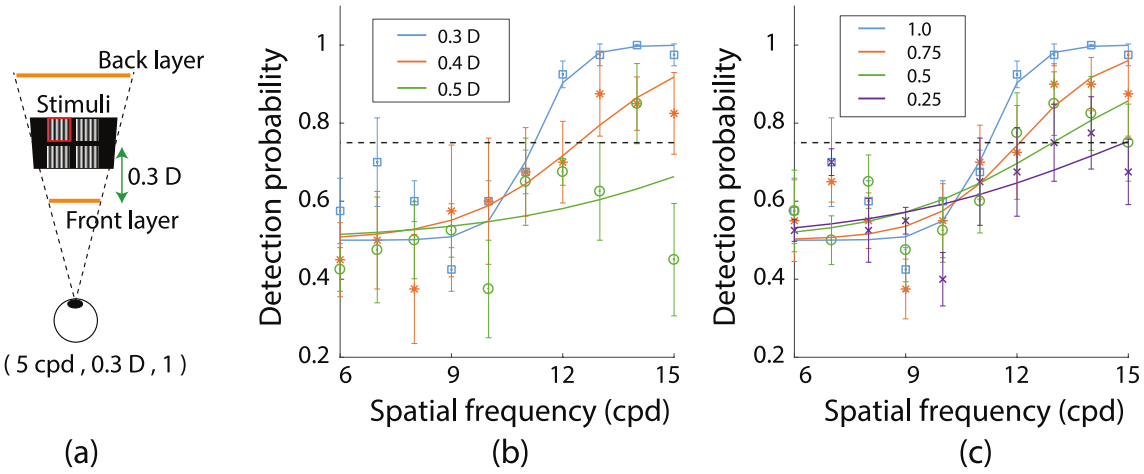


Figure 4.6: (a) The setup used for the texture distinguishability experiment. (b) The detection probability for different depth values with the contrast level set to 1, and (c) for different contrast levels, with the depth set to 0.3 D. Marked points show the mean scores across participants, and whiskers signify the standard deviation. Lines represent fitted psychometric functions to the data points. Dotted lines mark the 75% detection probability, considered a detection threshold (reused image from own publication [87])

The results of the experiment are displayed in Figure 4.6, (b-c). Lines represent the sigmoidal psychometric functions [81] that were fitted to the obtained data. The results confirm that the quality of LB at higher spatial frequencies decreases [54], as the detection threshold gets lower at higher cpd values. This decrease in detection can be observed for lower depth values and increased scene contrast.

SSIM-based comparison of LFS vs LB techniques

The experiments allowed drawing initial conclusions about the general difference between LB and LFS decomposition quality. However, used stimuli do not deliver complete information about their distinct features. An SSIM [75] was used to measure the similarity between various images to resolve this issue.

The LFS and LB methods were compared by computing the SSIM on them directly. The minimum value in the SSIM map acquired in the previous step was considered a *dissimilarity index*. The images from the experimental procedure in Section 4.3.1 were tested to confirm the findings. Apart from them, additional images were used to get new detection predictions. The results are shown in Figure 4.7. Please note that lower values indicate a larger difference from the ground truth.

The structural difference between images is small for most test cases. The main differences come from the images with higher spatial frequency patterns and depth set close to the middle. That follows the experiment results (Figure 4.6, (b)). Looking at the difference between Figure 4.7, (a) and Figure 4.7, (b) also shows

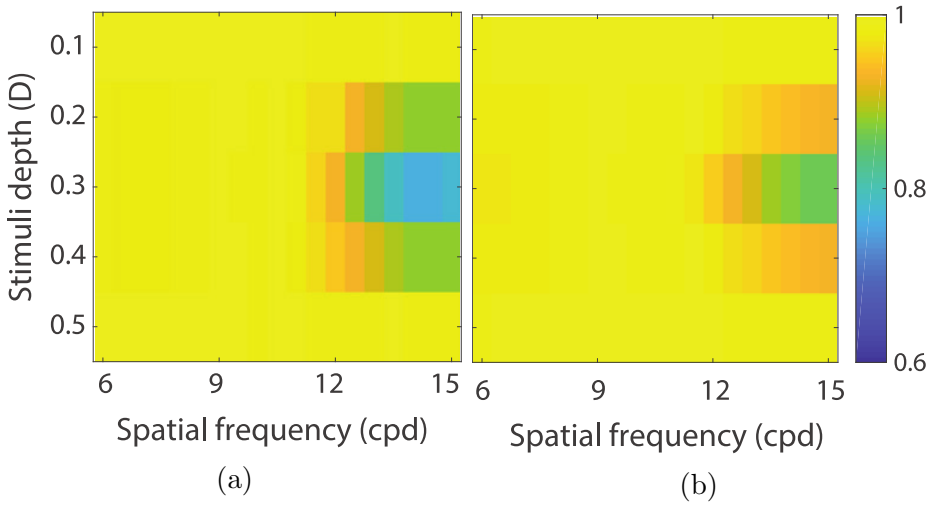


Figure 4.7: Dissimilarity index between LFS and LB-generated images (i.e. the minimum value in the SSIM map) with various spatial frequencies and depth values at contrast = 1 (a), and contrast = 0.5 (b) (reused image from own publication [87]).

that decreasing contrast lowers dissimilarity. That is also in accordance to the experiments (Figure 4.6, (c)).

Detection threshold. The most critical data gathered from the experiments is the detection threshold (see Figure 4.6). Images above this threshold are perceptually identical to the ground truth; those below have detectable differences. The minimum spatial frequency can be determined for every set depth and contrast parameter. A difference is considered detectable if its detection probability equals at least 75%. In order to generalise the findings across a broader range of stimuli, an SSIM-based dissimilarity index would have to be used. To this end, a root-mean-square (RMS) error was computed between the measured frequency threshold and various SSIM frequency thresholds. The frequency threshold set to 0.9 achieves the lowest error, becoming the general solution. Figure 4.8 shows the plot for all considered frequency thresholds.

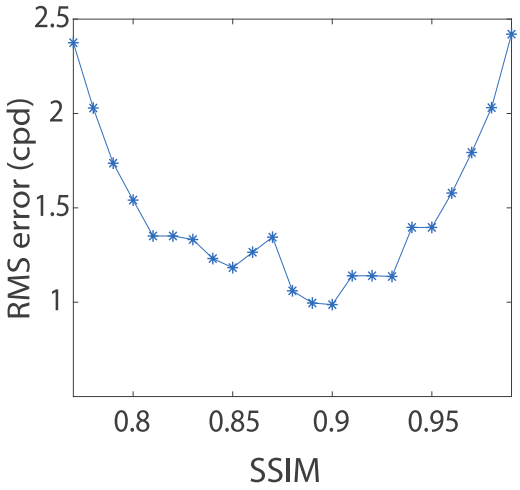
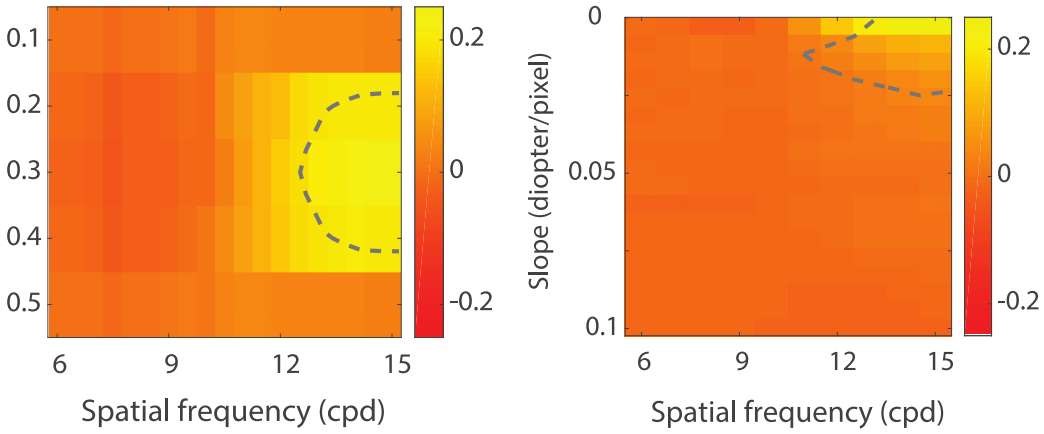


Figure 4.8: RMS error between experiment results and SSIM-based dissimilarity index based on the selected frequency threshold (reused image from own publication [87]).



(a) (b)

Figure 4.9: (a) The difference between dissimilarity indices for LFS and LB methods based on the ground truth data. Values above 0 indicate that LB is closer to the ground truth. Values below 0 show the higher quality of LFS. The region surrounded by a dashed line is the region where the difference between LFS and LB was detected by the participants of the experiment (see Figure 4.7). (b) The dissimilarity for slanted surfaces at different slopes (reused image from own publication [87]).

Ground truth. A similar analysis was performed using ground truth data by computing the SSIM map between LB and ground truth images and between LFS and ground truth. The resulting dissimilarity indices were compared to estimate the quality of the proposed methods. The results of this comparison are presented in Figure 4.9, (a).

As expected, LFS gives results closest to the ground truth in most tested conditions. LB performs better at the highest spatial frequencies and middle stimuli depths. However, according to the previous results, analysed methods are indistinguishable in most cases, which is represented by the points on the outside of the dashed line. The only region that is perceptually different depending on the chosen method (the region inside the dashed line) is in its entirety closer to the ground truth for LB. Therefore, LB is either indistinguishable from LFS or is closer to the ground truth. As a result, the LB method can be used for any textured regions without affecting the decomposition quality.

Slanted surfaces. The next step in measuring the perceived quality of textures is the analysis of slanted surfaces. This type of content is much more diverse than flat surfaces. Therefore the experimental procedure is even more challenging to create. Instead, the established SSIM-based metric is used and extended to slanted surfaces.

Because the rendered objects might span across the area between front and back planes, a simple SSIM map cannot be computed. It can, however, be calculated for different depths separately. To this end, seven focal images were created spanning

across the visual field (between 1.4 D and 2.0 D), with a step of 0.1 D. As in the flat surface case, the SSIM map is computed between a method and ground truth, this time for every focal plane separately. Then, after determining the minimum and obtaining the dissimilarity metric, the minimum value is calculated across all surfaces.

Eighteen slopes were tested, ranging from 0.005 D/pixel to 0.1 D/pixel. Slopes higher than 0.1D/pixel would span across less than 6 pixels on the 0.6 D visual field. Therefore, such slopes are treated as occlusion boundaries. The results of the tests are shown in Figure 4.9, (b). It can be seen that LFS is closer to the ground truth if LFS and LB are indistinguishable. The detectable difference (area surrounded by a dashed line, top right corner) favours LB similarly to flat surfaces in the cases when it is detectable. Therefore, it further confirms the previous assessment – LB is a better method for decomposition textures than LFS and can be used in any textured region.

Conclusions

The experiments have shown that, in most cases, the LFS and LB methods are indistinguishable. Even though some of the tested stimuli appear to be closer to the ground truth for LFS than LB, the difference is minuscule and not perceptible. When the methods are distinguishable, the LB achieves better results for every test. Therefore, it can render any textured region without any negative impact.

4.3.2 Depth discontinuity perception experiment

As previously established, LFS achieves higher quality than LB in the occlusion boundaries areas [89]. However, it is unknown if this statement holds in all possible situations involving varying depth differences, luminance contrasts, and eccentricities. Another experiment was performed that accounted for mentioned aspects to gather insight into depth discontinuities.

Perceptual experiment

The experimental procedure was based on the QUEST method [79]. The stimulus was shown in the same pattern as in the previous experiment (Section 4.3.1) for various luminance contrasts and eccentricities. Setting the luminance contrast involved manipulating the luminance of the back and front planes. The participants were asked to look at the green cross on the screen. The position of the cross changed for different eccentricities. The QUEST procedure optimised depth value – the distance between the front and back layers. The depth threshold was a distance for which

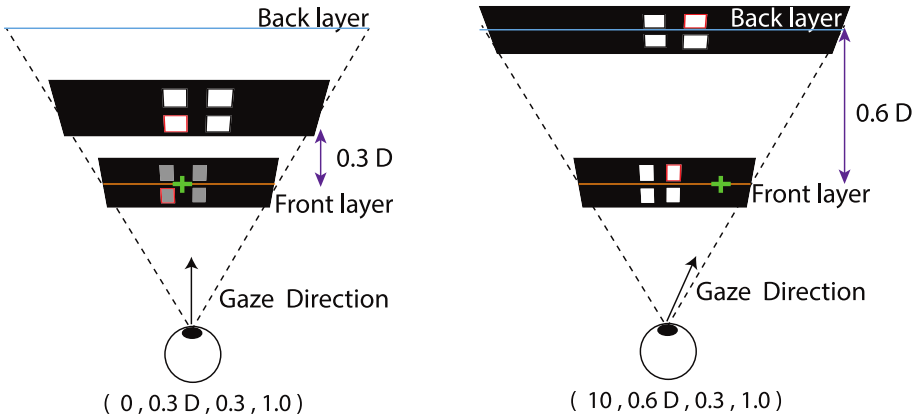


Figure 4.10: Exemplar stimuli used during the depth experiment. Two presented configurations are described by values in the parenthesis below the scheme. They indicate, from left to right: eccentricity (in visual degrees), depth, luminance of front layer, and luminance of back layer (reused image from own publication [87]).

LFS and LB are indistinguishable. During the procedure, depth adjusted itself until reaching the threshold. The value of depth ranged between 0.05 D and 0.6 D, with a step of 0.05 D. Examples of experiment stimuli are shown in Figure 4.10.

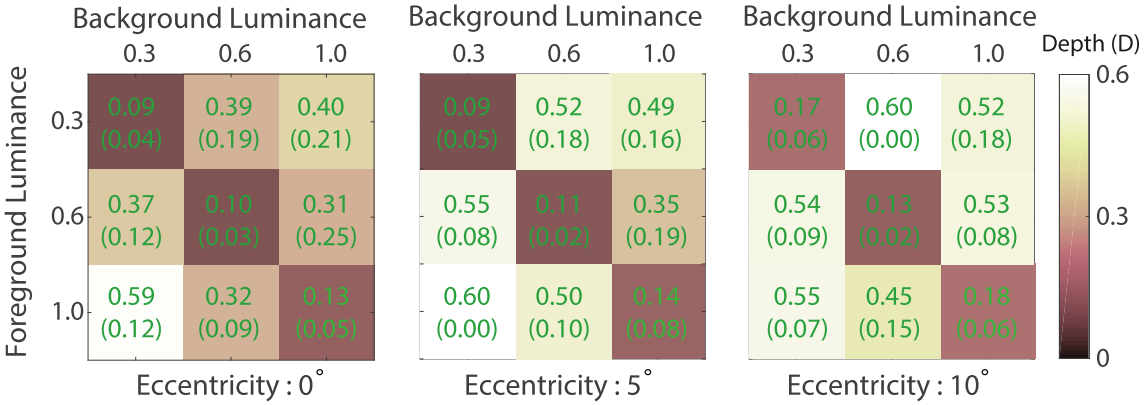


Figure 4.11: The results of the experiments showing the threshold depth for LFS and LB to be distinguishable. Values inside the squares indicate the mean threshold depth across all participants. Values in parenthesis show the standard deviation (reused image from own publication [87]).

Figure 4.11 contains the experiment results. Two main observations can be made. First of all, increasing the eccentricity increases the visibility threshold. This phenomenon allows replacing LFS with LB in the occlusions at the periphery without decreasing the quality. The second observation is that the distinguishability increases with lowered luminance contrast. It is opposite to what was discovered in the texture experiment (Figure 4.6, (b-c)). The analysis of this result is provided in Section 6.2 of the paper [87].

SSIM-base analysis

Similarly to the texture experiment, an SSIM-based metric was created to generalise findings. The detection rate could then be established for any given luminance values and eccentricities outside the tested range. The focal images were generated from the front plane perspective to calibrate the metric. The procedure was repeated for all tested luminance contrasts, eccentricities, and depth values. The images were blurred according to the human cut-off frequency values [78] to simulate the loss of visual acuity in the periphery. The foveal region (i.e. area around the gaze) was always kept at full quality. The Gaussian blur gradually increased with eccentricity.

After generating the images, the SSIM map was computed between LFS and LB versions. The minimum value of the map was selected as the dissimilarity index. Different detection thresholds were tested to find one with the lowest RMS error, similarly to the previous study. The lowest depth that fell below the chosen threshold was selected for each depth range. Figure 4.12 presents the prediction error for all cases.

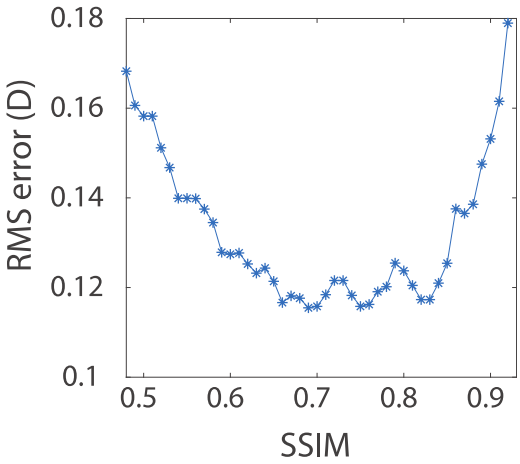


Figure 4.12: The RMS error calculated between values predicted using the SSIM metric and the experiment results (reused image from own publication [87]).

The threshold with the lowest error lies between 0.66 and 0.83. The maximum value in a range (0.83) was selected as the detection threshold to account for the potential inaccurate selection of LB instead of LFS. The difference between these prediction values and experimental results is presented in Figure 4.13. An SSIM generally overestimates the appropriate depth for settings where background luminance is lower than the foreground. On the other hand, the calculated depth value is lower than expected in the opposite case. The Michelson contrast for those two cases is the same. Therefore, the lowest values follow the conservative approach for such cases.

The SSIM metric for images outside the scope of the experimental procedure was calculated, as in the previous procedures. The images were generated with assumed eccentricity higher than 15°, up to 50°. A display with two additional virtual planes further back was simulated to account for depths higher than the separation between front and back planes (0.6 D). Such a simulation behaves the same way as the actual display for depths lower than 0.6 D; therefore, initial findings follow the additional performed tests.

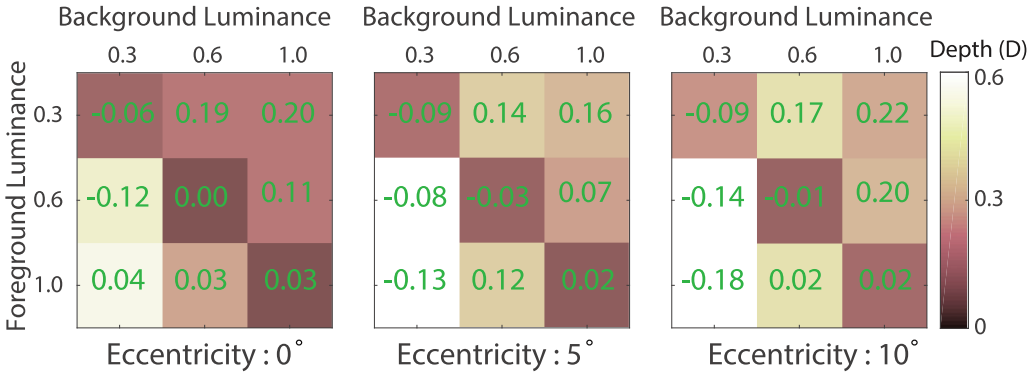


Figure 4.13: The difference between predicted SSIM depths with detectable difference and experimental results (reused image from own publication [87]).

Conclusions

Although using the LB creates occlusion artefacts, they are not always detectable. The experiment results showed the limitations of human vision in this regard. A function for calculating the maximum depth difference between the objects without visible distortions was created using contrast and eccentricity values. Thanks to it, the LFS method can be replaced by LB in some instances to improve the performance.

4.4 Integration of LFS and LB methods

Using the results obtained from the experiments and SSIM calibration, a selection rule can be established, deciding on the best method to use in any given situation. In terms of texture surfaces, the LB is proven to be sufficient. In the occlusion areas, the selection depends on various parameters. A model is proposed to create a general selection rule that considers all variables.

The goal regarding the occlusion boundaries is to find the minimum depth at which the difference between LB and LFS becomes apparent. A function was created using the data from the experiments and simulations. Its value was the depth threshold calculated based on the Michelson contrast and eccentricity parameters. This function was created by fitting the data points to the 3D surface. The results of this fitting are presented in Figure 4.14. The formula for a fitted plane is as follows:

$$D = 0.11 + 0.02\ E + 0.79\ C - 0.005\ E \times C - 0.379\ C^2, \tag{4.1}$$

where D is the depth threshold, E is eccentricity, and C is Michelson contrast.

The eccentricity has a high impact on the threshold value. It suggests that

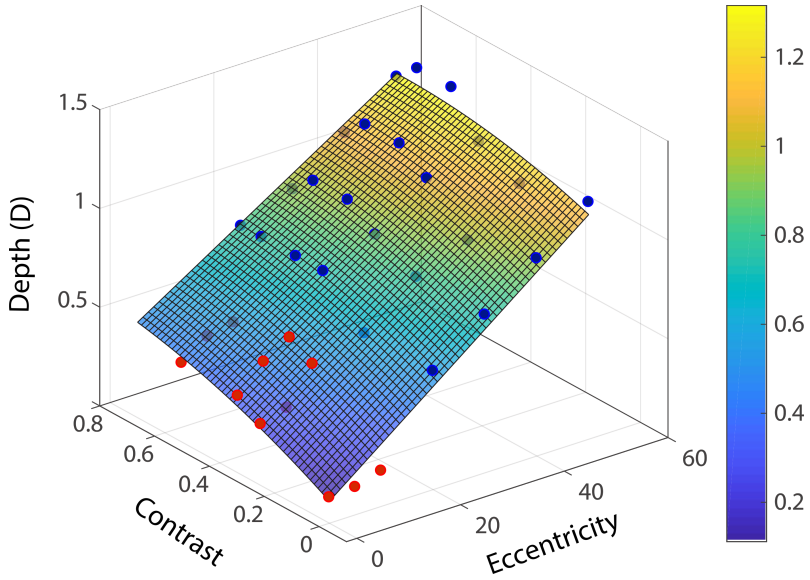


Figure 4.14: Predicted depth threshold for given Michelson contrast and eccentricity. Surface was fitted for data points taken from the experiments (red orbs) and data generated using SSIM metric (blue points) (reused image from own publication [87]).

using displays with a larger field of view might provide even higher gains.

The formula can be used directly in the hybrid optimisation procedure. For any point where the estimated depth threshold is higher than the value in the depth map, LB is employed. Otherwise, LFS is used. An example of a mask generated for all pixels is presented in Figure 4.15.

4.5 Implementation and method evaluation

The whole rendering procedure is defined in the following steps:

(a) (b) (c) (d)

Figure 4.15: Example of mask generation. (a) The initial view on the 3D scene. (b) The depth map used for LB. (c) Michelson contrast map used for estimating depth threshold. (d) Final generated mask. White pixels signify areas where LFS has to be performed according to the proposed model, assuming the user is looking at the centre. Green areas have to be masked additionally, if the eccentricity information is ignored (reused image from own publication [87]).

1. The scene is rendered from the central viewpoint using ray tracing based on the depth map. The size of the rendered image is 1200×1200 pixels. Each pixel is generated using one ray.

2. The LFS mask is computed using depth, contrast, and eccentricity maps.

3. Eight additional viewpoints are rendered in the masked regions around the central viewport.

4. Images are decomposed using LB and SART [4, 43] ran for a total of 10 iterations.

The model was implemented using the Nvidia OptiX ray tracer. The setup consists of four displays, beam splitters between two planes, and lenses that magnify the image. Two Pupil Labs eye-trackers are placed behind the lenses to track the central vision and measure eccentricity. A solution proposed by Kumar et al. [42] is used to stabilise and denoise the data gathered from them. The alignment of all components is presented in Figure 4.2.

4.5.1 Visual evaluation

An example of rendering is shown in Figure 4.16. The images were captured using a camera. Its focus was on selected depths between 1.4 D and 2.0 D. Additionally, the simulated versions of those captures were rendered. It was done by connecting the decomposed images back into one. Objects not in focus and at the periphery were blurred according to the cut-off frequency model [78] to simulate the human visual system. The simulation was then compared to the ground truth model. To this end, an SSIM map was created, using which the hybrid method, LB, and LFS were compared.

It can be observed that LB performs well in most tested conditions. However, as expected, it differs significantly from the ground truth in the occlusion areas; the edges of the occluding objects are very pronounced. LFS can deal with them without creating significant differences from the ground truth. However, it fails in some highly textured areas, as shown in the first forest scene example. The high-frequency regions are blurred, which decreases their visual quality. Overall, LB can create very sharp edges and transitions. LFS creates smooth transitions and, in general, evens out the surfaces. The proposed method combines those two approaches by creating smooth edges of the objects and sharp, textured regions. SSIM maps comparing the rendering with ground truth show that the hybrid method is the closest to the ideal light field.

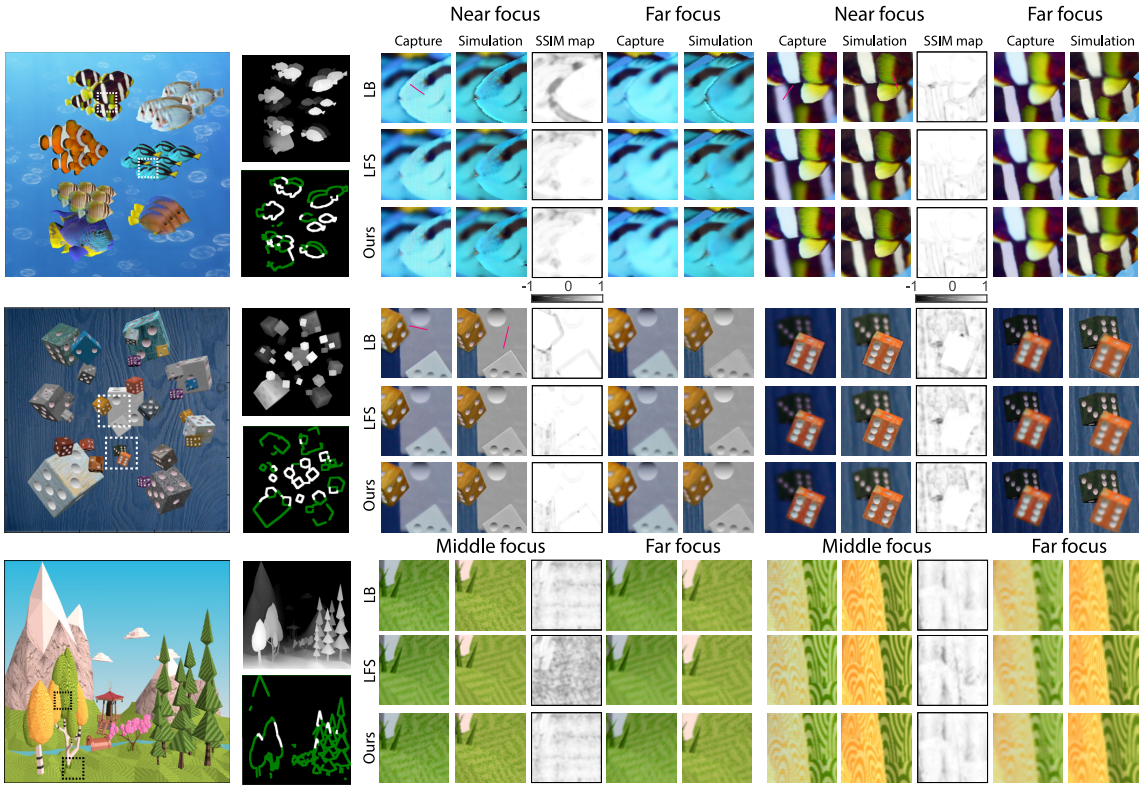


Figure 4.16: Three exemplar renderings using the proposed method. The first column shows the scenes, and the second column contains the depth and light field synthesis map. Additional images contain the visualisation from particular regions of the scenes at various focus depths. Each example consists of three images: image captured by the camera, simulated rendering of this image, and SSIM map between simulation and ground truth (reused image from own publication [87]).

Apart from static scenes, the temporal coherence of decomposition was also evaluated. As all textured regions were created using LB decomposition, they are consistent in the temporal domain – the artefacts might appear only at the occlusions where LFS and LB methods switch between each other in subsequent frames. However, as demonstrated in Section 4.3.2, the switch occurs only at the threshold point. Therefore, in such situations, both methods should be indistinguishable.

The dynamic scenes with translations and rotations applied to the objects were created to confirm the hypothesis. The videos are available as the supplementary material to the paper [88] showing the animations and LFS maps. The gaze position is assumed to be in the centre of the display. In the area close to the gaze centre, no distortions are visible – the transition between methods is smooth. In the peripheral regional artefacts appear occasionally. However, when observed from the appropriate eccentricity, they are not visible. The textured regions are generally temporally consistent, as expected. However, artefacts appear at the high spatial frequency regions in the first shown scene. They are caused by the sampling rate used in the ray tracing algorithm. This problem may be fixed by increasing the

Scene	Polygons	Masked	Rendering time	Decomposition time	Frame rate
Fish	20498	7.3%	9.26 (27.48) ms	2.57 (4.11) ms	85 (32) Hz
Dice	569810	6.5%	14.11 (47.08) ms	2.44 (4.12) ms	60 (20) Hz
Forest	16924	1.8%	7.29 (28.31) ms	2.35 (4.19) ms	104 (31) Hz

Table 4.1: Benchmark of the hybrid method for three scenes. Numbers in the last three columns signify its performance. Numbers in parenthesis measure the timings of generating full light fields (LFS).

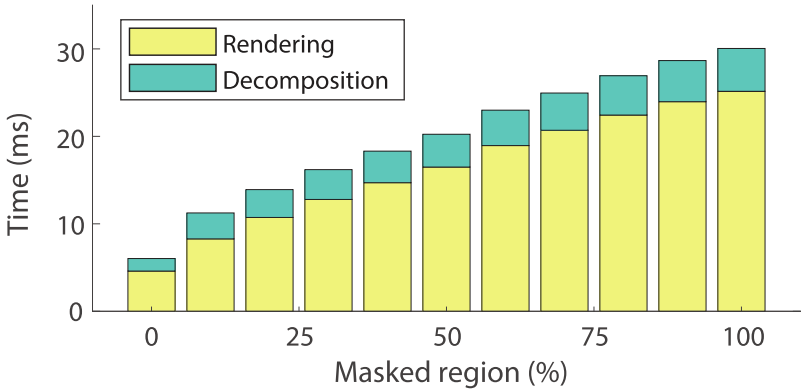


Figure 4.17: The performance of the hybrid method for randomly sampled LFS mask (reused image from own publication [87]).

number of rays cast per pixel.

In the third scene, the simulation with a mouse-controlled gaze is presented. The movement of the gaze point does not introduce any spatial artefacts.

4.5.2 Performance

The performance of the proposed method was measured and compared to the LFS. The results are presented in Table 4.1. The hybrid method achieves significantly higher performance in all tested cases. Another performance test was created to confirm the linear impact on the frame time of the mask coverage. The mask was sampled randomly for the selected pixel fraction between 0% and 100%. For each of those tests, the decomposition and rendering time was measured. The results are presented in Figure 4.17. It shows that the performance scales linearly with the number of samples. It is also worth noting that the time of LFS rendering is not equal to LB rendering time scaled by the number of viewpoints. It is due to the performed joint optimisation of LFS and LB. More details on this framework are provided in Section 7.2 of the paper [87].

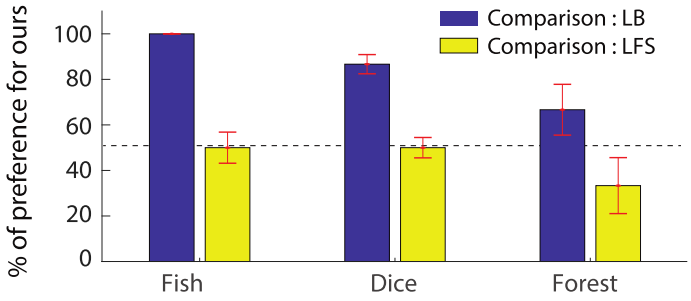


Figure 4.18: The preference of hybrid method when compared to LB and LFS. Values indicate at what percentage of cases the hybrid method was preferred when compared with another one. Whiskers show the standard deviation across all participants (reused image from own publication [87]).

4.5.3 Experiment

A perceptual experiment was created to evaluate the hybrid method formally. In several tasks, the hybrid decomposition quality was compared to the LFS and LB. In each task, the participant could switch freely between two methods. One of the methods was hybrid, and the second was either LFS or LB. The order of stimuli was random. The participants had to look at the specific point of the screen to prevent the generation of new LFS images. Had the participant moved their gaze position outside the $2^\circ \times 2^\circ$ box, the stimulus was replaced with the grey screen until the gaze returned inside. Three scenes were used, the same as shown in Figure 4.16. For each of them, five different gaze locations were tested. Six participants took part in the experiment. Its results are presented in Figure 4.18.

The results show that the participants preferred the hybrid method significantly more than LB in all the scenes ($p < 0.05$ for all). When compared with LFS, the results are mixed. In the fish and dice scenes, the results show mixed selections. None of the methods has a statistical preference. It is a satisfactory conclusion as the hybrid method achieves similar perceptual quality as a much more expensive method. However, the participants preferred LFS more in the forest scene ($p < 0.05$). They have reported that blurred textures looked better than sharp ones generated using hybrid decomposition. This impression might be due to too precise reconstruction of textures. Because of small head movements, inaccuracies in pupil detection, and calibration errors, high spatial frequencies are perceived as noise.

4.6 Chapter summary

This Chapter described a hybrid decomposition method composed of LB and LFS. Multiple perceptual experiments were performed in which both approaches were tested thoroughly. For content with high spatial frequencies, LB is distinguishable from LFS and is also closer to the ground truth. For all lower spatial frequencies, the methods are indistinguishable. LFS performs better for occlusion boundaries, but there are cases in which it is perceptually identical to LB. By combining those statements, a new approach was created. It achieves very high quality while decreasing the performance cost significantly.

This work, however, comes with a few limitations. Most of them are associated with hardware constraints. The main inaccuracies of the hybrid model come from the fact that the build was equipped with magnifying lenses. It creates aberrations in the outer regions of the image, limiting the space to place content. Other limitations come from the size of the display. Although it has a relatively large field of view when compared to similar near-eye multi-focal displays (40°) [40, 47], it is 2-3 times lower than that of currently available VR goggles. The depth of produced content is also heavily narrowed, as the stimuli can only be modelled between 1.4 D and 2.0 D.

Another limitation comes from a model of depth discontinuity perception. Since there is no established perception-based metric, the model is based on the SSIM. It does not account for such variables as head movements, sensitivity to peripheral vision, or accommodation. Developing metrics that include the analysis of light fields and display parameters would be beneficial for this study.

Author's contributions. The decomposition method described in this Chapter was part of the project realised by the research team at the Max Planck Institute in Saarbruecken, Germany. This project is published in a joined paper by Yu et al. [87]. I participated in the regular meetings, where we shared the ideas and progress made on the project. I was particularly responsible for the rendering implementation, which used the decomposition algorithms at its core, and scenes preparation for experiments. I also implemented the eye-tracking components and ran the perceptual experiments that involved its usage. I participated in the discussion on the results of the experiments, and I was responsible for refining the software and procedures according to the conclusions made during the project meetings.

I was not engaged in activities related to the design and construction of the multi-focal display hardware nor in the implementation of the decomposition algorithm.

Chapter 5

Summary

This work presents methods of using perception to generate images for stereoscopic displays. These methods use the human visual system features to speed up existing methods and improve the perceptual quality of the image they generate.

Foveated rendering, in which the image sampling varies based on the image content, is used to reduce the calculation time by a factor of two compared to standard methods. It is done without any visible reduction in image quality.

The foveated rendering can also be extended by the adaptation to light. It is shown that mapping tones can simulate visual adaptation using colour perception on a virtual reality display. The temporal aspect of adaptation was tested, to determine the best performing model. Overall, the experiments show that users prefer the simplified adaptation model compared to the realistic one.

To combat the issue of temporal aliasing present in the foveated rendering, an aliasing detection procedure has been shown. The method is presented that allows obtaining information about image elements particularly exposed to the appearance of aliasing caused by low image resolution.

A novel method of creating images using GAN architecture from a limited number of samples was proposed. Contrary to the existing methods, the focus was on creating an image that was not an exact copy of the original but only its indistinguishable metamer. This approach allowed the constraints imposed on the neural network by restrictive loss functions to be lowered. It resulted in a reduction of the computation costs with an increase in the quality of the created image.

Existing light-field synthesis methods make it possible to create high-quality images with high computational costs. Conversely, faster methods based on simple linear blending have significant inaccuracies at the edges of the objects. A hybrid method combining both approaches was presented. It allowed obtaining the maximum image quality with a three times lower rendering cost than light field synthesis.

Conclusions

In this work, the following conclusions have been made:

- The information about the scene content and eccentricity can be used to differentiate the rendering rates for every image fragment without affecting its quality (Section 2.2, published in [71]).
- The visual adaptation of the human visual system can be simulated on the VR headset with limited dynamic range. The studies show that participants prefer a simplified model more than the realistic one that is based on photoreceptor responses to changing light conditions (Section 2.3, published in [80]).
- It is possible to obtain the dataset viable for measuring visual perception of temporal aliasing artefacts. The dataset can be further used for neural network training to establish the rule for selecting the image areas that are affected by it the most (Section 2.4).
- Neural networks can be trained to generate metamers instead of full-resolution images. Such a method reduces the requirements posed on training and inference time and increases the quality of results (Chapter 3).
- For multi-focal displays, the complex light field synthesis method can be replaced in certain parts of the image by a simpler method of linear blending. A hybrid method was created that can calculate which method is sufficient for any image fragment so that the final image is of the highest visual quality (Chapter 4).

Reference to the thesis

In the introduction, the thesis of the dissertation was formulated as follows:

Considering the perceptual features of the human visual system improves the efficiency of the synthesis of stereoscopic images while maintaining a comparable perceptual quality of these images.

During the research on this topic, various methods have been proposed to increase the speed of generating stereoscopic images designed for VR goggles and multi-focal displays. The foveated rendering and hybrid image decomposition showed an increase in rendering speed without impacting the image quality.

Additionally, new approaches were shown to increase the rendering quality. The visual adaptation increases the quality of reproduced colours on display in

spatial and temporal domains. A new method of GAN training to create metamers results in higher output quality than standard input.

Therefore, in this dissertation, the formulated thesis has been confirmed.

Future work

The topics explained in this thesis can be directly continued in the future. Firstly, the introduced foveation method considers the spatial information on the scene only. The contrast sensitivity could also be measured in the temporal domain by including subsequent frames in the predictor calculation. Such an aspect of vision could lead to further improvements in performance.

The topic of visual adaptation also has additional aspects that can be expanded in the future. As the human visual system consists of three cone types, they can be accounted for differently. Additional perception effects can also be included, e.g. the afterimages appearing as bright blobs after being exposed to large luminance. Lastly, the content-aware foveation predictor could be extended to include the adaptation process in the calculation. It would make it possible to join both systems into one.

The proposed method of estimating the location of noticeable aliasing requires a more considerable amount of data for proper neural network training. In future, this data could be gathered by evaluating a more diverse aliasing dataset with a larger group of participants included in the calibration experiment. By extending the input in such a way, the trained network could be improved to assess the aliasing quality with much higher quality. The trained network could then be integrated into the rendering system to accurately measure its impact on the performance compared to existing anti-aliasing methods.

The idea of creating metamers using neural networks is a novelty that could be expanded in many possible ways. For example, instead of training the GAN on images with specific JND differences compared to full resolution, it might also be beneficial to use images that have a lower level of distortions. New distortions types could also be introduced to limit them to constrained texture synthesis only. Furthermore, currently used separate networks for near and far peripheries could be unified to create a general solution for creating a complete image in one inference.

Multi-focal displays rely mostly on the hardware, which is still only in a prototype form. Created hybrid decomposition method relies on a specific, custom-made display with limited resolution and field of view. In the future, a more advanced display can be built. It could be used to create a more robust decomposition model, generalized for any multi-focal display.

List of Figures

List of Tables

Own publications

- Surace, L., Wernikowski, M., Tursun, O., Myszkowski, K., Mantiuk, R. and Didyk, P., 2021. *Learning Foveated Reconstruction to Preserve Perceived Image Statistics*. arXiv preprint arXiv:2108.03499.

- Tursun, O.T., Arabadzhiyska-Koleva, E., Wernikowski, M., Mantiuk, R., Seidel, H.P., Myszkowski, K. and Didyk, P., 2019. *Luminance-contrast-aware foveated rendering.* ACM Transactions on Graphics (TOG), 38(4), pp.1-14.

- Yu, H., Bemana, M., Wernikowski, M., Chwesiuk, M., Tursun, O.T., Singh, G., Myszkowski, K., Mantiuk, R., Seidel, H.P. and Didyk, P., 2019. *A perception-driven hybrid decomposition for multi-layer accommodative displays.* IEEE transactions on visualization and computer graphics, 25(5), pp.1940-1950.

- Wernikowski, M., Mantiuk, R. and Piórkowski, R., 2019, January. *Preferred Model of Adaptation to Dark for Virtual Reality Headsets.* In International Conference on Multimedia Modeling (pp. 118-129). Springer, Cham.

- Wernikowski, M., 2016. *Preferred Speed of Visual Adaptation to Darkness in Computer Games.* Central European Seminar on Computer Graphics.

Bibliography

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

[2] Kurt Akeley, Simon J Watt, Ahna Reza Girshick, and Martin S Banks. A stereo display prototype with multiple focal distances. *ACM transactions on graphics (TOG)*, 23(3):804–813, 2004.

[3] AMD FidelityFX™ Super Resolution. `https://www.amd.com/en/technologies/radeon-software-fidelityfx-super-resolution`, 2021.

[4] Anders H Andersen and Avinash C Kak. Simultaneous algebraic reconstruction technique (sart): a superior implementation of the art algorithm. *Ultrasonic imaging*, 6(1):81–94, 1984.

[5] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.

[6] Peter GJ Barten. 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*, volume 1077, pages 73–82. International Society for Optics and Photonics, 1989.

[7] Mary Ann Branch, Thomas F Coleman, and Yuying Li. A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems. *SIAM Journal on Scientific Computing*, 21(1):1–23, 1999.

[8] B. G. Breitmeyer and H. Ogmen. Visual masking. *Scholarpedia*, 2(7):3330, 2007. revision #182339.

[9] Andrew Burnes. Grand Theft Auto V PC Graphics & Performance Guide. `https://www.nvidia.com/en-us/geforce/news/grand-theft-auto-v-pc-graphics-and-performance-guide`, 2015.

[10] Andrew Burnes. NVIDIA DLSS 2.0: A Big Leap In AI Rendering. `https://www.nvidia.com/en-us/geforce/news/nvidia-dlss-2-0-a-big-leap-in-ai-rendering`, 2020.

[11] Peter J Burt and Edward H Adelson. The laplacian pyramid as a compact image code. In *Readings in computer vision*, pages 671–679. Elsevier, 1987.

[12] Fergus W Campbell and John G Robson. Application of fourier analysis to the visibility of gratings. *The Journal of physiology*, 197(3):551–566, 1968.

[13] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3606–3613, 2014.

[14] Christine A Curcio, Kimberly A Allen, Kenneth R Sloan, Connie L Lerea, James B Hurley, Ingrid B Klock, and Ann H Milam. Distribution and morphology of human cone photoreceptors stained with anti-blue opsin. *Journal of comparative neurology*, 312(4):610–624, 1991.

[15] Hugh Davson. *Physiology of the Eye*. Macmillan International Higher Education, 1990.

[16] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[17] Serge O Dumoulin and Brian A Wandell. Population receptive field estimates in human visual cortex. *Neuroimage*, 39(2):647–660, 2008.

[18] James A Ferwerda, Sumanta N Pattanaik, Peter Shirley, and Donald P Greenberg. A model of visual adaptation for realistic image synthesis. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 249–258, 1996.

[19] Jeremy Freeman and Eero P Simoncelli. Metamers of the ventral stream. *Nature neuroscience*, 14(9):1195–1201, 2011.

[20] Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. *Advances in neural information processing systems*, 28:262–270, 2015.

[21] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[22] Steven J Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F Cohen. The lumigraph. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 43–54, 1996.

[23] Brian Guenter, Mark Finch, Steven Drucker, Desney Tan, and John Snyder. Foveated 3d graphics. *ACM Transactions on Graphics (TOG)*, 31(6):1–10, 2012.

[24] Alexander Hepburn, Valero Laparra, Ryan McConville, and Raul Santos-Rodriguez. Enforcing perceptual consistency on generative adversarial networks by using the normalised laplacian pyramid distance. *arXiv preprint arXiv:1908.04347*, 2019.

[25] David M Hoffman, Ahna R Girshick, Kurt Akeley, and Martin S Banks. Vergence–accommodation conflicts hinder visual performance and cause visual fatigue. *Journal of vision*, 8(3):33–33, 2008.

[26] Fu-Chung Huang, David P Luebke, and Gordon Wetzstein. The light field stereoscope. In *SIGGRAPH emerging technologies*, pages 24–1, 2015.

[27] Robert William Gainer Hunt. *The reproduction of colour*, volume 4. Wiley Online Library, 1995.

[28] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[29] Jorge Jimenez, Jose I Echevarria, Tiago Sousa, and Diego Gutierrez. Smaa: enhanced subpixel morphological antialiasing. In *Computer Graphics Forum*, volume 31, pages 355–364. Wiley Online Library, 2012.

[30] Jorge Jimenez, Diego Gutierrez, Jason Yang, Alexander Reshetov, Pete Demoreuille, Tobias Berghoff, Cedric Perthuis, Henry Yu, Morgan McGuire, Timothy Lottes, et al. Filtering approaches for real-time anti-aliasing. *SIGGRAPH Courses*, 2(3):4, 2011.

[31] Jost B Jonas, Ulrike Schneider, and Gottfried OH Naumann. Count and density of human retinal photoreceptors. *Graefe's Archive for Clinical and Experimental Ophthalmology*, 230(6):505–510, 1992.

[32] Eric R Kandel, James H Schwartz, Thomas M Jessell, Steven Siegelbaum, A James Hudspeth, and Sarah Mack. *Principles of neural science*, volume 4. McGraw-hill New York, 2000.

[33] Anton S Kaplanyan, Anton Sochenov, Thomas Leimkühler, Mikhail Okunev, Todd Goodall, and Gizem Rufo. Deepfovea: neural reconstruction for foveated rendering and video compression using learned statistics of natural videos. *ACM Transactions on Graphics (TOG)*, 38(6):1–13, 2019.

[34] Armin Kappeler, Seunghwan Yoo, Qiqin Dai, and Aggelos K Katsaggelos. Video super-resolution with convolutional neural networks. *IEEE transactions on computational imaging*, 2(2):109–122, 2016.

[35] Brian Karis. High-quality temporal supersampling. *Advances in Real-Time Rendering in Games, SIGGRAPH Courses*, 1(10.1145):2614028–2615455, 2014.

[36] Shrutik Katchhi and Pritish Sachdeva. A review paper on oculus rift. *International Journal of Current Engineering and Technology E-ISSN*, pages 2277–4106, 2014.

[37] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[38] Vamsi Kiran Adhikarla, Marek Vinkler, Denis Sumin, Rafal K Mantiuk, Karol Myszkowski, Hans-Peter Seidel, and Piotr Didyk. Towards a quality metric for dense light fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 58–67, 2017.

[39] Julius William Kling. *Woodworth and Schlosberg's experimental psychology*. Holt Rinehart and Winston, 2021.

[40] Robert Konrad, Emily A Cooper, and Gordon Wetzstein. Novel optical configurations for virtual reality: Evaluating user preference and performance with focus-tunable and monovision near-eye displays. In *Proceedings of the 2016 CHI conference on human factors in computing systems*, pages 1211–1220, 2016.

[41] Jonathan Korein and Norman Badler. Temporal anti-aliasing in computer generated animation. In *Proceedings of the 10th annual conference on Computer graphics and interactive techniques*, pages 377–388, 1983.

[42] Manu Kumar, Jeff Klingner, Rohan Puranik, Terry Winograd, and Andreas Paepcke. Improving the accuracy of gaze input for interaction. In *Proceedings*

of the 2008 symposium on Eye tracking research & applications, pages 65–68, 2008.

[43] Douglas Lanman, Gordon Wetzstein, Matthew Hirsch, Wolfgang Heidrich, and Ramesh Raskar. Polarization fields: dynamic light field display using multi-layer lcds. In *Proceedings of the 2011 SIGGRAPH Asia Conference*, pages 1–10, 2011.

[44] Seungjae Lee, Changwon Jang, Seokil Moon, Jaebum Cho, and Byoungho Lee. Additive light field displays: realization of augmented reality with holographic optical elements. *ACM Transactions on Graphics (TOG)*, 35(4):1–13, 2016.

[45] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, 2(2):164–168, 1944.

[46] Marc Levoy and Pat Hanrahan. Light field rendering. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 31–42, 1996.

[47] Patrick Llull, Noah Bedard, Wanmin Wu, Ivana Tosic, Kathrin Berkner, and Nikhil Balram. Design and optimization of a near-eye multifocal display system for augmented reality. In *Propagation through and Characterization of Distributed Volume Turbulence and Atmospheric Phenomena*, pages JTH3A–5. Optical Society of America, 2015.

[48] Thurmon E Lockhart and Wen Shi. Effects of age on dynamic accommodation. *Ergonomics*, 53(7):892–903, 2010.

[49] Lester Loschky, George McConkie, Jian Yang, and Michael Miller. The limits of visual resolution in natural scene viewing. *Visual Cognition*, 12(6):1057–1092, 2005.

[50] Kevin J MacKenzie, David M Hoffman, and Simon J Watt. Accommodation to multiple-focal-plane displays: Implications for improving stereoscopic displays and for accommodation control. *Journal of vision*, 10(8):22–22, 2010.

[51] Radosław Mantiuk and Sebastian Janus. Gaze-dependent ambient occlusion. In *International Symposium on Visual Computing*, pages 523–532. Springer, 2012.

[52] Rafał K Mantiuk, Anna Tomaszewska, and Radosław Mantiuk. Comparison of four subjective methods for image quality assessment. In *Computer graphics forum*, volume 31, pages 2478–2491. Wiley Online Library, 2012.

[53] Olivier Mercier, Yusufu Sulai, Kevin Mackenzie, Marina Zannoli, James Hillis, Derek Nowrouzezahrai, and Douglas Lanman. Fast gaze-contingent optimal decompositions for multifocal displays. *ACM Transactions on Graphics (TOG)*, 36(6):1–15, 2017.

[54] Rahul Narain, Rachel A Albert, Abdullah Bulbul, Gregory J Ward, Martin S Banks, and James F O'Brien. Optimal presentation of imagery with focus cues on multi-plane displays. *ACM Transactions on Graphics (TOG)*, 34(4):1–12, 2015.

[55] Nvidia. VRWorks - Variable Rate Shading (VRS) website. `https://developer.nvidia.com/vrworks/graphics/variablerateshading`, 2018. Accessed: 2021-09-06.

[56] G. Osterberg. *Topography of the Layer of Rods and Cones in the Human Retina*. Acta ophthalmologica. Supplementum. A. Busck, 1935.

[57] Stephen E Palmer. *Vision science: Photons to phenomenology*. MIT press, 1999.

[58] Anjul Patney, Marco Salvi, Joohwan Kim, Anton Kaplanyan, Chris Wyman, Nir Benty, David Luebke, and Aaron Lefohn. Towards foveated rendering for gaze-tracked virtual reality. *ACM Transactions on Graphics (TOG)*, 35(6):1–12, 2016.

[59] Eli Peli. Contrast in complex images. *JOSA A*, 7(10):2032–2040, 1990.

[60] Eli Peli, Jian Yang, and Robert B Goldstein. Image invariance with changes in size: The role of peripheral contrast thresholds. *JOSA A*, 8(11):1762–1774, 1991.

[61] Erik Reinhard, Wolfgang Heidrich, Paul Debevec, Sumanta Pattanaik, Greg Ward, and Karol Myszkowski. *High dynamic range imaging: acquisition, display, and image-based lighting*. Morgan Kaufmann, 2010.

[62] FJ Richards. A flexible growth function for empirical use. *Journal of experimental Botany*, 10(2):290–301, 1959.

[63] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[64] Louise Ryana, Kevin J MacKenziea, and Simon J Watta. Multiple-focal-planes 3d displays: A practical solution to the vergence-accommodation conflict? In *2012 International Conference on 3D Imaging (IC3D)*, pages 1–6. IEEE, 2012.

[65] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[66] John L Spudich, Chii-Shen Yang, Kwang-Hwan Jung, and Elena N Spudich. Retinylidene proteins: structures and functions from archaea to humans. *Annual review of cell and developmental biology*, 16(1):365–392, 2000.

[67] Michael Stengel, Steve Grogorick, Martin Eisemann, and Marcus Magnor. Adaptive image-space sampling for gaze-contingent real-time rendering. *Computer Graphics Forum*, 35(4):129–139, 2016.

[68] Luca Surace, Marek Wernikowski, Okan Tursun, Karol Myszkowski, Radosław Mantiuk, and Piotr Didyk. Learning foveated reconstruction to preserve perceived image statistics. *arXiv preprint arXiv:2108.03499*, 2021.

[69] Xin Tao, Hongyun Gao, Renjie Liao, Jue Wang, and Jiaya Jia. Detail-revealing deep video super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4472–4480, 2017.

[70] LN Thibos, FE Cheney, and DJ Walsh. Retinal limits to the detection and resolution of gratings. *JOSA A*, 4(8):1524–1529, 1987.

[71] Okan Tarhan Tursun, Elena Arabadzhiyska-Koleva, Marek Wernikowski, Radosław Mantiuk, Hans-Peter Seidel, Karol Myszkowski, and Piotr Didyk. Luminance-contrast-aware foveated rendering. *ACM Transactions on Graphics (TOG)*, 38(4):1–14, 2019.

[72] Robert A Ulichney. Void-and-cluster method for dither array generation. In *Human Vision, Visual Processing, and Digital Display IV*, volume 1913, pages 332–343. International Society for Optics and Photonics, 1993.

[73] Brian Wandell and Stephen Thomas. Foundations of vision. *Psyccritiques*, 42(7), 1997.

[74] Zhou Wang and Alan C Bovik. Modern image quality assessment. *Synthesis Lectures on Image, Video, and Multimedia Processing*, 2(1):1–156, 2006.

[75] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[76] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pages 1398–1402. Ieee, 2003.

[77] Greg Ward. A contrast-based scalefactor for luminance display. *Graphics Gems*, 4:415–21, 1994.

[78] Andrew B Watson and Albert J Ahumada. Blur clarified: A review and synthesis of blur discrimination. *Journal of Vision*, 11(5):10–10, 2011.

[79] Andrew B Watson and Denis G Pelli. Quest: A bayesian adaptive psychometric method. *Perception & psychophysics*, 33(2):113–120, 1983.

[80] Marek Wernikowski, Radosław Mantiuk, and Rafał Piórkowski. Preferred model of adaptation to dark for virtual reality headsets. In *International Conference on Multimedia Modeling*, pages 118–129. Springer, 2019.

[81] Felix A Wichmann and N Jeremy Hill. The psychometric function: I. fitting, sampling, and goodness of fit. *Perception & psychophysics*, 63(8):1293–1313, 2001.

[82] Krzysztof Wolski, Daniele Giunchi, Nanyang Ye, Piotr Didyk, Karol Myszkowski, Radosław Mantiuk, Hans-Peter Seidel, Anthony Steed, and Rafał K Mantiuk. Dataset and metrics for predicting local visible differences. *ACM Transactions on Graphics (TOG)*, 37(5):1–14, 2018.

[83] Kai Xiao, Gabor Liktor, and Karthik Vaidyanathan. Coarse pixel shading with temporal supersampling. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 1–7, 2018.

[84] Lei Xiao, Salah Nouri, Matt Chapman, Alexander Fix, Douglas Lanman, and Anton Kaplanyan. Neural supersampling for real-time rendering. *ACM Transactions on Graphics (TOG)*, 39(4):142–1, 2020.

[85] Lei Yang, Shiqiu Liu, and Marco Salvi. A survey of temporal antialiasing techniques. In *Computer Graphics Forum*, volume 39, pages 607–621. Wiley Online Library, 2020.

[86] Lei Yang, Diego Nehab, Pedro V Sander, Pitchaya Sitthi-Amorn, Jason Lawrence, and Hugues Hoppe. Amortized supersampling. *ACM Transactions on Graphics (TOG)*, 28(5):1–12, 2009.

[87] Hyeonseung Yu, Mojtaba Bemana, Marek Wernikowski, Michał Chwesiuk, Okan Tarhan Tursun, Gurprit Singh, Karol Myszkowski, Radosław Mantiuk, Hans-Peter Seidel, and Piotr Didyk. A perception-driven hybrid decomposition for multi-layer accommodative displays. *IEEE transactions on visualization and computer graphics*, 25(5):1940–1950, 2019.

[88] Hyeonseung Yu, Mojtaba Bemana, Marek Wernikowski, Michał Chwesiuk, Okan Tarhan Tursun, Gurprit Singh, Karol Myszkowski, Radosław Mantiuk, Hans-Peter Seidel, and Piotr Didyk. Temporal coherence of hybrid-decomposition. `http://lfd.mpi-inf.mpg.de/files/lfd_video.mp4`, 2019.

[89] Marina Zannoli, Gordon D Love, Rahul Narain, and Martin S Banks. Blur and the perception of depth at occlusions. *Journal of Vision*, 16(6):17–17, 2016.

[90] Wenjun Zeng, Scott Daly, and Shawmin Lei. Point-wise extended visual masking for jpeg-2000 image compression. In *Proceedings 2000 International Conference on Image Processing (Cat. No. 00CH37101)*, volume 1, pages 657–660. IEEE, 2000.

[91] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[92] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.