

Mitigating Visual Discomfort on Head Mounted Displays using Estimated Gaze Dependent Depth of Field

by

Kieran Carnegie

A thesis
submitted to the Victoria University of Wellington
in partial fulfilment of the
requirements for the degree of
Master of Science
in Computer Science.

Victoria University of Wellington
2015

Abstract

Virtual Reality (VR) applications on Head Mounted Displays (HMDs) are now more common and accessible for personal viewing than before with the introduction of consumer-level devices like the Oculus Rift. However, exposure to VR applications on HMDs results in significant discomfort for the majority of people, the severity of which can both increase or decrease after repeated exposures. This is disadvantageous for the development and adoption of VR, as a long adaptation period cannot be relied on for making Virtual Environments palatable.

Symptoms of discomfort caused by the viewing of content on VR devices including HMDs are historically described as “Simulator Sickness” and include eye fatigue, headaches, nausea and sweating; symptoms very similar to those experienced by sufferers of motion sickness. We refer to the specific subset of Simulator Sickness Symptoms caused by visual stimuli as symptoms of “Visual Discomfort”.

A conflict between accommodation and vergence depth cues on stereoscopic displays is known to be a significant cause of visual discomfort. This report describes a psychophysical evaluation used for judging the effectiveness of dynamic Depth of Field (DoF) blurring on reducing visual discomfort caused by initial exposure to stereoscopic content on HMDs.

Our DoF implementation adjusts the focal region of stereoscopic content based on an estimation of users’ view vectors in real time and is realised in a commercial game engine. Participants report a significant reduction of visual discomfort using a simulator sickness questionnaire when DoF blurring is enabled. On average, a 34% reduction in our sickness measure is observed, indicating that dynamic DoF blurring is an effective rendering technique for reducing visual discomfort.

Acknowledgements

I would like to express gratitude to my supervisor, Dr Taehyun Rhee, for allowing me to work on such an interesting topic, and for the guidance and support he has offered through my masters.

My thanks to my father, Dale Carnegie, for his immense help and support, this thesis wouldn't be anywhere near as good without him.

My sincere thanks to my friends, particularly Till Elton, Jay Flint and Emma Greenbank, for their willingness to put up with my crazy ideas and sit down with strange contraptions strapped to them to see what happens.

Finally, I would like to acknowledge my family and friends who have supported me throughout my time at university.

Contents

Title Page	
Abstract	i
Acknowledgements	ii
Glossary	vii
List of Figures	x
List of Tables	xiii
List of Algorithms	xv
1 Introduction	1
1.1 Research Motivation	1
1.2 Research Objectives	3
1.3 Project Methodology	4
1.4 Thesis Structure	5
2 Previous Research and Theory	7
2.1 HMDs	7
2.2 An Introduction to Simulator Sickness	10
2.3 General Causes of Simulator Sickness	11
2.3.1 Vection	12
2.3.2 Expectation mismatch	13
2.4 Causes of simulator sickness specific to HMDs	14
2.4.1 The vestibular system and vestibulo-ocular reflex	14
2.4.2 Accommodation-vergence conflict	18
2.4.3 Immersion and abnormal motions	22
2.4.4 Hardware considerations	22
2.5 Reducing Simulator Sickness	25
2.5.1 Hardware solutions	25
2.5.2 Software solutions	26
2.5.3 Using DoF blur to reduce simulator sickness	28

3	Estimated Gaze Dependent DoF	31
3.1	Overview of the EGD-DoF Algorithm	32
3.2	Gaze estimation	33
3.2.1	Estimating the user's gaze direction	33
3.2.2	Finding the focal depth	33
3.2.3	Updating the focal depth	36
3.2.4	Gaze estimation summary	37
3.3	Real Time Depth of Field	38
3.4	Selecting the Blur Algorithm	40
4	Software Implementation	45
4.1	Unreal Engine v4	45
4.1.1	Blueprints and the UE4 editor	46
4.2	Overview of the EGD-DoF implementation	48
4.3	Implementing Gaze Estimation	49
4.4	Implementing DoF blur	60
4.5	Interacting with the scenes	63
4.6	Selecting DoF Parameters and Constants	65
4.7	Implementation Summary	66
5	Psychophysical Evaluation	67
5.1	Setup	67
5.1.1	Participants	67
5.1.2	Testing hardware	69
5.1.3	Testing scenes	69
5.2	Stimuli	70
5.3	Procedure	73
5.4	The Metric for Determining Visual Discomfort	76
6	Results and Analysis	81
6.1	Withdrawals	81
6.2	Individual Symptom Results and Analysis	84

<i>CONTENTS</i>	vii
6.3 Independent Variables	87
6.4 Discomfort Categories	89
6.5 Overall Visual Discomfort	91
6.6 Summary of Results	93
7 Discussion and Conclusion	95
7.1 Summary	95
7.2 Limitations and Future Work	97
7.3 Conclusion	98
A Overview of the EGD-DoF system	111
B Scenes used in this research	115
C Forms and documents used in the Study	119

List of Acronyms

3D	Three Dimensional
2D	Two Dimensional
AR	Augmented Reality
CoC	Circle of Confusion
CRT	Cathode Ray Tube
DFoV	Display Field of View
DoF	Depth of Field
EGD	Estimated-gaze-dependent
EGD-DoF	Estimated-gaze-dependent Depth of Field
FoV	Field of View
GFoV	Geometric Field of View
GPU	Graphics Processing Unit
HMD	Head Mounted Display
LCD	Liquid Crystal Display
PI-VR	Personally-Immersive Virtual Reality
SDK	Source Development Kit
SSQ	Simulator Sickness Questionnaire
VAR	Variable acuity resolution
VE	Virtual Environment
VOR	Vestibulo-ocular Reflex
VR	Virtual Reality

List of Figures

1.1	The Oculus Rift DK1.	2
2.1	Comparison of modern and historical HMD design patents .	8
2.2	Examples of current generation HMDs and their classifica- tions	9
2.3	An optokinetic drum, used to induce vection.	13
2.4	The vestibular system.	15
2.5	The vestibulo-ocular reflex.	16
2.6	Vision correcting glasses require a different VOR ratio. . . .	17
2.7	Overview of human depth cues.	18
2.8	Accommodation as a depth cue.	19
2.9	Vergence as a depth cue.	20
2.10	The accommodation-vergence conflict.	21
2.11	An orientation-independent (fixed) background reduces vi- sual discomfort.	27
3.1	Overview of the EGD-DoF system.	32
3.2	In order to avoid focusing at unlikely depths, multiple rays are cast into the scene to find the optimal focal depth.	34
3.3	Overview of the focal depth estimation algorithm.	38
3.4	The effect of bokeh DoF on a sample photograph.	41
3.5	Example of bokeh and Gaussian DoF in UE4.	42
3.6	Overview of a basic implementation of bokeh DoF.	44
4.1	An explanatory UE4 blueprint, highlighting each of the dif- ferent types of components used in this research.	49
4.2	Finding the position and rotation of the player's camera. . .	51
4.3	Setting the target point for rays cast into the scene.	52
4.4	Casting a ray into the scene.	53

4.5	Casting eight rays at small angular deviations into the scene.	54
4.6	Setting the interpolation speed.	57
4.7	The blueprint that controls the interpolation to a new focal depth.	58
4.8	Focal distance interpolation.	59
4.9	The post-process node that implements bokeh blur.	59
4.10	Approximation of the blur radius for different depth disparities output by the UE4 bokeh implementation.	62
4.11	Example of a UE4 scene with differing radii of DoF blur. . .	62
4.12	The player avatar, with attached camera.	63
4.13	Test scene used for selecting DoF parameters	66
5.1	A participant during the user study.	70
5.2	Sun Temple scene sample screenshots.	71
5.3	Landscape Mountains scene sample screenshots.	72
6.1	Frequency distribution of S_T scores.	82
6.2	Frequency distribution of S_C scores.	83
6.3	S_T results for each session, ordered by the date the session was completed.	84
6.4	Average change in symptom severity, for symptoms used in S_T calculation.	85
6.5	Average change in symptom severity, for symptoms not used in S_T calculation.	86
6.6	S_T scores, comparing EGD-DoF enabled/disabled.	88
6.7	S_T scores, comparing participant's first and second sessions.	88
6.9	Total sickness score for participants with (left) and without (right) DoF enabled.	91
6.8	Change of sickness measure for each of the three discomfort categories: Nausea (top), Oculomotor (centre), Disorientation (bottom).	92

LIST OF FIGURES

xiii

A.1	UE4 Blueprint: The EGD-DoF system (first half).	112
A.2	UE4 Blueprint: The EGD-DoF system (second half).	113
B.1	Sun Temple render	116
B.2	Landscape Mountains render	117

List of Tables

2.1	Types of HMD, using AR categories from [1, 2]	9
4.1	Parameters for the UE4 bokeh function	60
5.1	Conditions that excluded participation in the study.	68
5.2	Comparison of factors in each scene	70
5.3	SSQ symptoms and the corresponding discomfort categories	75
5.4	A fictitious participant's SSQ results and S_T score.	79
6.1	Comparison of mean discomfort across each of the discomfort categories.	89

List of Algorithms

3.1	Construct rays for finding the optimal depth for user focus .	35
3.2	Calculate d_f	36
3.3	Estimating the bokeh effect using postprocessing	42

Chapter 1

Introduction

1.1 Research Motivation

In 2012, Oculus VR Inc. was founded and successfully had their Head Mounted Display (HMD), the Oculus Rift DK1 (the Oculus) funded on Kickstarter [3]. This made widely available, for the first time, a personal VR viewing device at a price point suitable for consumers. Established companies have begun making and modifying applications specifically for VR, such as Valve Software adding VR modes to multiple games including Half Life 2 and Team Fortress 2 [4]. The openly available Oculus Source Development Kit (SDK) also allows developers to trivially create their own VR applications, so VR development is not just limited to large studios. This has resulted in a rapidly growing ecosystem of applications, games and films appropriate for VR.

HMDs theoretically represent an ideal system for the viewing of VR applications (also referred to as Virtual Environments (VEs)): their immersive nature creates a strong presence illusion, where users perceive the VE as real and not mediated through technology [5]. The major practical issue with HMDs lies in the fact that users commonly report adverse physical reactions when using them, including discomfort symptoms such as headaches, nausea, dizziness and eye-strain [6]. Collectively, these symptoms represent a condition termed “simulator sickness” [7]. It has been found that simulator sickness is experienced by up to 80% of users of HMDs [8]. Preliminary investigations among University students and staff shows this figure to be higher. The specific subset of these symptoms



Figure 1.1: The Oculus Rift DK1. Image from en.wikipedia.org

caused by visual stimuli is hereafter referred to as “visual discomfort”.

Repeated exposure to VEs on HMDs is known to reduce the severity and incidence of visual discomfort [9, 10] for some, but not all users [11]: some users find that repeated exposure actually results in a negative feedback loop where more exposure increases their discomfort. It is impractical to rely on this acclimatisation period for reducing discomfort, as it is unlikely that normal consumers will willingly expose themselves to sickening stimulation in the uncertain hope that it “gets better”.

A solution that addresses discomfort experiences during a user’s first HMD exposure is thus essential to the continued growth and adoption of VR.

1.2 Research Objectives

The primary objective of this research is to identify a solution that mitigates visual discomfort when viewing stereoscopic content on HMDs. Optimally, the solution will meet the following requirements:

- It will not require any specific hardware beyond that available in unmodified, current generation, consumer-appropriate HMDs. It should also require no physical alteration to the HMDs, excepting trivial modifications such as repositioning lenses.
- Hardware solutions fragment the market, making it unreasonable to expect content developers to produce content that is optimised across multiple devices using significantly differing technologies. A software solution is thus optimal, especially if it can be included in SDKs such as the Oculus': this would remove the burden of implementation from content producers.
- It should be applicable to a wide variety of 3D applications without unnecessarily restricting content producers.
- It should involve no degradation of user experience or control in VEs.
- It should be testable without the requirement for external hardware, as such external hardware can be intrusive in psychophysical evaluations of consumer devices and may not accurately represent viewing conditions outside the lab.

1.3 Project Methodology

The thesis objectives require the following steps to be completed:

- Investigate the behaviours of the vestibular and ocular systems, HMD design factors, and how they interact.
- Identify elements of the design of HMDs that cause discomfort when viewing stereoscopic content.
- Identify a software-only solution to address these causes of discomfort.
- Create or obtain scenes usable for refining the solution and perceptually evaluating its effectiveness.
- Implement the solution, preferably in a commercial game engine.
- Using the above scenes;
 - Perform a psychophysical evaluation of user discomfort with and without the solution enabled.
- Evaluate the effectiveness of the solution in reducing visual discomfort.

1.4 Thesis Structure

The thesis is presented as follows:

- Chapter 2** summarises prior research into visual discomfort in HMDs. The technologies used in current generation HMDs are introduced, along with simulator sickness and both its general and HMD specific causes. Finally, the hardware and software systems employed in previous research to address visual discomfort are investigated.
- Chapter 3** presents a estimated-gaze-dependent depth of field (EGD-DoF) algorithm, which is proposed to reduce visual discomfort when viewing stereoscopic content on HMDs.
- Chapter 4** presents an implementation of the EGD-DoF system in a commercial game engine.
- Chapter 5** overviews the set-up and methodology of a psychophysical evaluation designed to assess the effectiveness of the EGD-DoF system at reducing visual discomfort. The two 3D immersive scenes used to evaluate the effectiveness of this system are also presented. The change in sickness score, S_T is introduced as a metric for evaluating changes in visual discomfort.
- Chapter 6** presents and analyses the results gathered from the psychophysical evaluation. A Wilcoxon signed-rank test is performed to evaluate the significance of these results. Detailed discussions on specific aspects and conclusions drawn from the data are also presented.
- Chapter 7** concludes the thesis with a reflection on the objectives of the research, recommendations for future work and a summary of the results obtained during this research.

Chapter 2

Previous Research and Theory

2.1 HMDs

In 1960 a patent for a “Stereoscopic-television apparatus for individual use” was awarded to Heilig Morton L [12]. The description offered for such devices read as follows:

“A stereoscopic television apparatus for individual use comprising an enclosed, hollow casing, one portion of said casing being concave to fit about the face of a wearer, said one portion having two eye openings, means to mount said casing on the head of a wearer with said eye openings in position to be looked through by the wearer, [with] two television units mounted within said casing so that one television unit is visible through each of said eye openings...”

Or, in plain English: “A display device, worn either by itself on the head or as part of a helmet, that brings an optical device directly in front of a user’s eyes so it may be viewed at small distance.” This 1960 patent was the first formal description of a device that can be retroactively recognised as a HMD, and interestingly is fundamentally identical to the design of current generation HMDs (Figure 2.1).

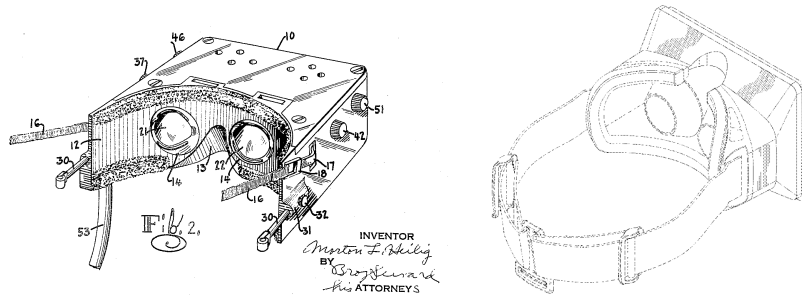


Figure 2.1: Diagram from L’s patent for a HMD, 1960 (left) compared to a diagram from Oculus VR Inc.’s Virtual Reality patent, 2014 (right).

In 1961, Philco Corporation produced a device called the “Headsight Surveillance Unit”, which is generally agreed to be the first HMD produced [13, 14]. The Headsight system was able to display content from remote closed-circuit cameras, but was unable to display virtual content: it was not until 1968 that a HMD was produced that could. This HMD, called the “Sword of Damocles”¹ [15] also tracked the roll, pitch and yaw of a user’s neck motions, translating them into rotations of the camera within the VE, a behaviour also present in the significantly more recent Oculus Rift DK1 [16]. Translational motion is also captured in the successor to the DK1, the Oculus Rift DK2, albeit with the added inconvenience of requiring an external tracking camera [17].

Many differing categorisation systems for identifying devices used for viewing VEs have been proposed, such as those in [18, 19]. These systems often focus on different design elements for distinguishing between categories, and no standardised system yet exists. Throughout this literature review, two device classifications were consistently encountered that detailed HMD designs. In this research, these two categories are referred to as Augmented reality (AR) and personally-immersive VR (PI-VR). In gen-

¹Named for its significant bulk, which required mounting to the roof above a user’s head.

eral, an AR HMD allows for a real-world view with digital modification and/or augmentation, and a PI-VR device occludes all external stimuli, displaying only a virtual image. Table 2.1 details a summary of the different subtypes of both AR and PI-VR displays, and Figure 2.2 shows examples of some current generation HMDs and their classifications.

Table 2.1: Types of HMD, using AR categories from [1, 2]

Type of HMD		Characteristics
AR	Feed-through	A camera on the front of the HMD supplies a visual feed of reality which is then augmented.
	Partially occluding	Does not occlude the entirety of the user's visual field, allowing reality to be viewed alongside the screen.
	Translucent	The HMD screen is translucent, allowing reality to be viewed through the screen.
PI-VR	Binocular	Differing screens and/or images are shown to each eye
	Biocular	Each eye views the same screen.



Figure 2.2: Examples of current generation HMDs and their classifications. From left to right: Google glass (Translucent AR), Oculus Rift DK1 (Binocular PI-VR), Sony HMZ-T2 (Binocular PI-VR)

Of specific interest to this research are binocular PI-VR displays, as the majority of current research and development into HMDs involve this category of device. In binocular HMDs, the images shown to each eye typically involve a small angular discrepancy, creating the impression of depth via stereopsis. Content rendered in this manner is called “stereoscopic”. In comparison, monoscopic content refers to content with no angular disparity (i.e. the images displayed to each eye are identical). The Oculus that is used in this study is a PI-VR HMD, and is typically used to display stereoscopic content [16].

PI-VR HMDs provide a level of presence that cannot be equalled by other display systems [20]. The result of achieving presence is referred to as immersion or as a place illusion (defined as “the sensation of being in a real place”) in a purely virtual scene [21]. The benefits of achieving high levels of immersion include improvements in learning (when compared to learning tasks on traditional desktop systems) [22], as well as allowing psychological conditioning such as the treating of phobias that would not otherwise be possible without physical stimulus [23]. However, achieving optimal levels of presence in VEs is known to require precisely calibrated scenes and optimised hardware, the currently accepted specifications of which are discussed in Section 2.4.4.

2.2 An Introduction to Simulator Sickness

In the previous section it was mentioned that devices that are unable to meet optimal rendering thresholds for human perception will not achieve total immersion for users. This “imperfect immersion” is not the only downfall: users of these devices also develop sickness symptoms that resemble those of motion sickness [8]. The incidence of such symptoms was first noticed in 1958, before the first HMD patent was even filed,

on a helicopter training simulator called the “2-F2-H hover trainer” [24]. Trainees using the simulator commonly reported a series of adverse reactions during their exposure. A catch-all definition of simulator sickness was coined to describe these phenomena: “the diverse signs or symptoms that have been experienced by flight crews during or after a training session in a flight simulator” [7].

A more specific definition of simulator sickness is provided by the US army: “discomfort [occurring] in a simulator of any kind” [25]. These definitions do not make any distinction based on the cause of the sickness symptoms, only that they arose in a “simulator”. For this reason, this research introduces the term “visual discomfort”, specifically to refer to simulator sickness caused by purely visual stimuli.

It is accepted that the symptoms of such discomfort can be split into three categories [26]: (1) oculomotor symptoms such as eye-strain, headaches and blurry vision, (2) disorientation symptoms such as dizziness and vertigo, and (3) nausea symptoms such as changes in salivation and stomach awareness. A single symptom may belong to multiple categories: blurry vision for example, is considered a component of both oculomotor and disorientation discomfort.

2.3 General Causes of Simulator Sickness

Early research into simulator sickness was performed before the first HMDs were actually built and focussed heavily on the hardware designs of the VR systems available at the time. For example, it was suggested in 1958 (three years before Philco’s HMD was produced) that specific hardware components used in the 2-F2-H hover trainer such as low update frequency CRT monitors and helicopter motion bases were the causes of sim-

ulator sickness [24]. Identical symptoms also occur in PI-VR HMDs [27], indicating that the identified hardware components in these simulators was not the sole factor in causing simulator sickness. It is reasonable to conclude simulator sickness is intrinsically tied to immersion in VR systems of widely differing design, and not just to specific hardware. This section details some of these more general causes of simulator sickness.

2.3.1 Vection

Vection is defined as “the powerful illusion of self-motion” [28]. When a large part of the visual field moves, the viewer believes that they are moving and that the world around them is stationary [29]. This is related to the effect of virtual presence, where a user’s self-orientation and self-location become defined by the virtual world they are immersed in, instead of the physical world their body inhabits [30]. Hence, when a large portion of the visual field moves due to a change in the view in a simulator, the user experiences an implied motion of their self-location, creating the illusion of self-motion.

In a study using an optokinetic drum to induce vection (Figure 2.3), around 60% of participants reported motion sickness symptoms when subjected to a changing visual field while they remained stationary [31]. This result suggests that the existence of a conflict between the illusory perceived self-motion and the actual motions of a user is a cause of discomfort. This is supported by studies that show displays that induce effects such as vection are more likely to cause sickness when compared to displays that do not [28, 32].



Figure 2.3: An optokinetic drum. The drum's interior is vertically striped, so when the drum rotates, a person sitting inside the drum is subjected to a moving visual field while they remain stationary.

Another example of this conflict between perceived and actual self-motion occurs when reading a book in a car: the actual motion of moving in the car does not match the lack of motion perceived by reading the relatively motionless book. Experiencing such motion conflicts in a car is known to cause motion sickness [33]. It is thus proposed that conflicts between perceived and actual motions are a cause of visual discomfort.

2.3.2 Expectation mismatch

In the previous section, it was concluded that a conflict between perceived and actual motion is a cause of discomfort. Similarly, motions that do not match those we expect can be sickening (for example, when standing on the deck of a ship, the expected lack of motion conflicts with the ship's rocking motion in the water, causing seasickness) [34]. When pilots were using the hover trainer mentioned in Section 2.2, a correlation was observed between actual flight hours logged and the severity of the discomfort experienced as a result of exposure to the simulator [24]. It was

hypothesised that more experienced pilots better remembered the motions of a helicopter and were perturbed when the motions of the simulator did not exactly match their expectations. Less experienced pilots were yet to build up this expectation and as such, did not experience the same discomfort.

2.4 Causes of simulator sickness specific to HMDs

HMDs are known to cause the same vection and expectation mismatch issues that have been identified in other simulators [35]. The greater levels of immersion experienced in HMDs compared to other VR systems exacerbates these effects, typically leading to increased discomfort. This section details some of the specific physiological characteristics of humans that are known to cause issues when viewing stereoscopic content on HMDs, as well as some of the HMD specific causes of visual discomfort.

2.4.1 The vestibular system and vestibulo-ocular reflex

Components in the inner ear called “otoliths” and “semicircular canals” are respectively responsible for detecting linear and angular rotations of the head [36]. These are parts of the vestibular system, which is responsible for balance and spatial orientation in mammals. As part of this role, the vestibular system stabilises retinal images during head motions [37]. Figure 2.4 shows a simplified diagram of the vestibular system.

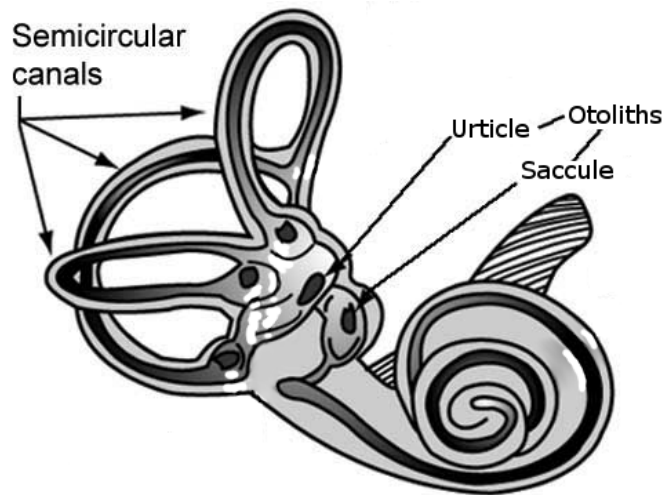


Figure 2.4: The vestibular system. Each of the three orthogonal semicircular canals contain fluid that moves during a rotation of the head, and each of the two otoliths detect linear accelerations through small hair cells. Modified from <http://weboflife.nasa.gov/learningResources/vestibularbrief.htm>

An interaction between the vestibular and ocular systems known as the Vestibulo-Ocular Reflex (VOR) is responsible for maintaining this retinal stability. Whenever the head either rotates or translates, the eyes rotate or translate around the same axis, in the opposing direction [38] (Figure 2.5). Ideally the ratio of eye to head movements, called the “VOR ratio” should be 1:1, as other ratios will cause retinal image slip in normal viewing situations [39]. However, through a process called VOR adaptation, different VOR ratios can be learnt.

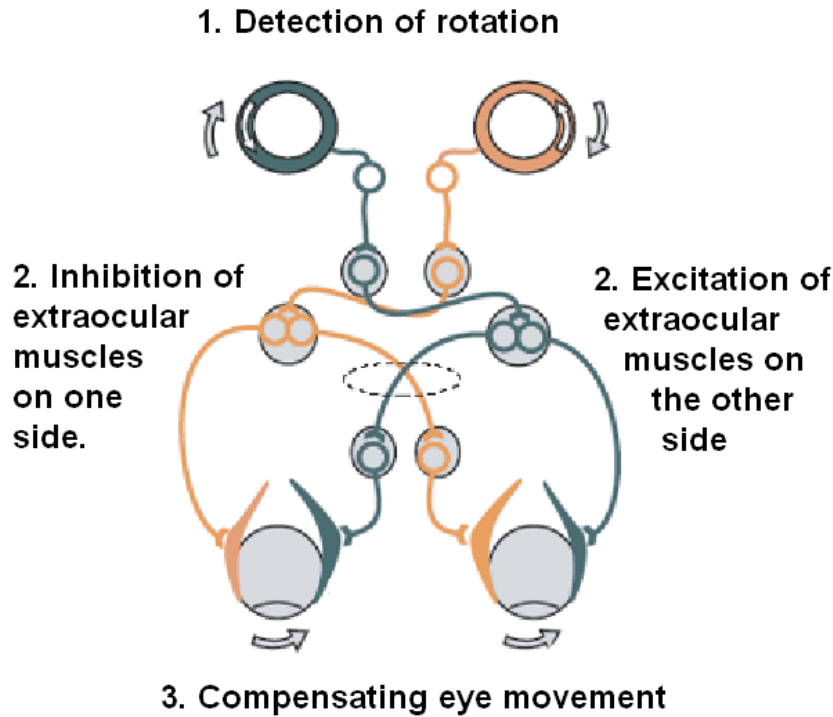


Figure 2.5: The vestibulo-ocular reflex, where a rotation detected in the semi-circular canals (bottom) results in opposing movement of the eyes (top). Image from en.wikipedia.org

An example of a situation where learning a new VOR ratio is required is when a person adjusts to a new vision correcting glasses prescription [40]. Light rays from blurry content will take up a larger portion of the visual field than when the same content is viewed with correct focus, so different eye motions are required depending on whether objects are blurry or not (Figure 2.6). This adaptation can occur within a few hours under optimal conditions and is context dependent [41]: the visual system can dynamically switch between different learnt VOR ratios as required.

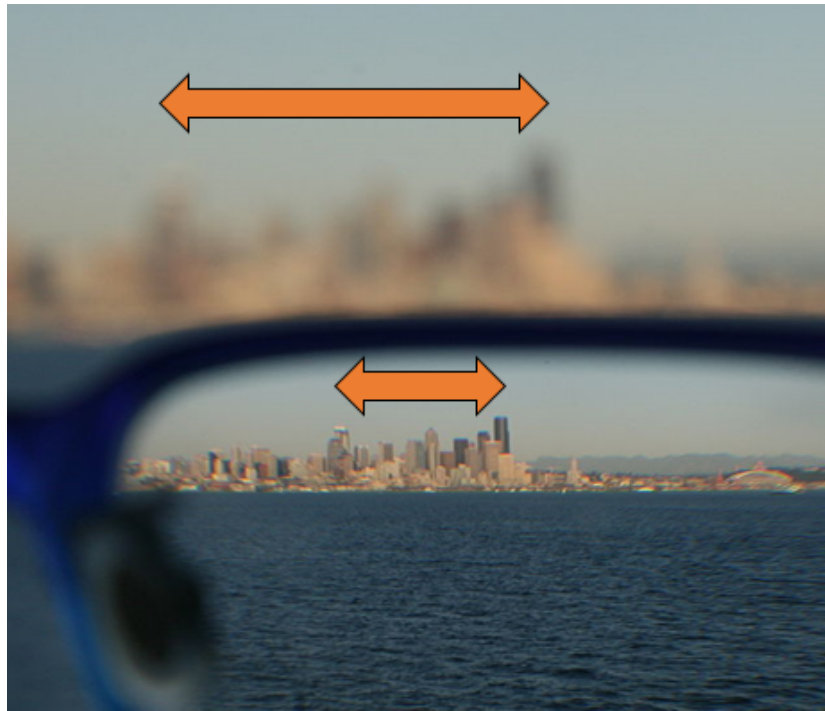


Figure 2.6: Vision correcting glasses require a different VOR ratio: Objects that are viewed through the glasses will require a smaller angular motion to look between when viewed through glasses (comparison of orange arrows). Image taken from [11]

The optics in HMDs can produce a VOR ratio different from 1:1, especially if the user has not accurately calibrated the system for their precise viewing conditions. Exposure to HMDs takes one to two weeks of continual use to fully learn the new VOR ratio if it does not precisely match expected values. During the period where the visual system is adapting to the new VOR ratio, discomfort is experienced due to the retinal slip caused by this mismatch between head movements and the resulting eye movements [11].

2.4.2 Accommodation-vergence conflict

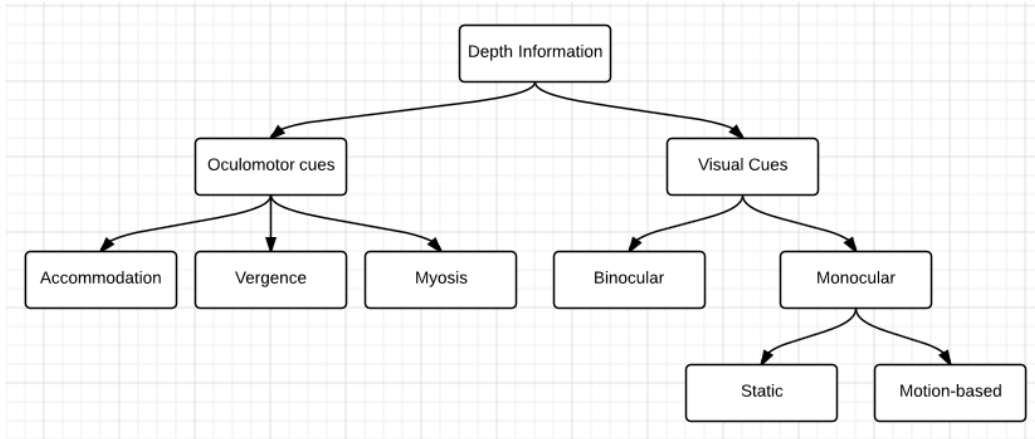


Figure 2.7: A summary of the cues used to infer depth by the human visual system.

The human visual system utilizes a variety of cues to infer depth [42] (Figure 2.7). These cues can be split into two broad categories: Visual depth cues that involve inferring depth from visual stimuli and Oculomotor depth cues that involve the physical responses of the ocular system in response to visual input.

Visual depth cues are either binocular or monocular, if they rely on visual stimulus to a single eye (monocular) or both eyes (binocular). Monocular depth cues can be classified depending on whether they are motion-based, such as parallax or dynamic occlusion; or static such as linear perspective, relative size and light and shadow distribution. Oculomotor depth cues can be separated into three categories, of which two are of interest in this research: accommodation and vergence.

Accommodation is the mechanism by which the eye alters its optical properties to bring objects at varying distances into focus, so that they form a sharp image on the retina (Figure 2.8). Objects outside the current

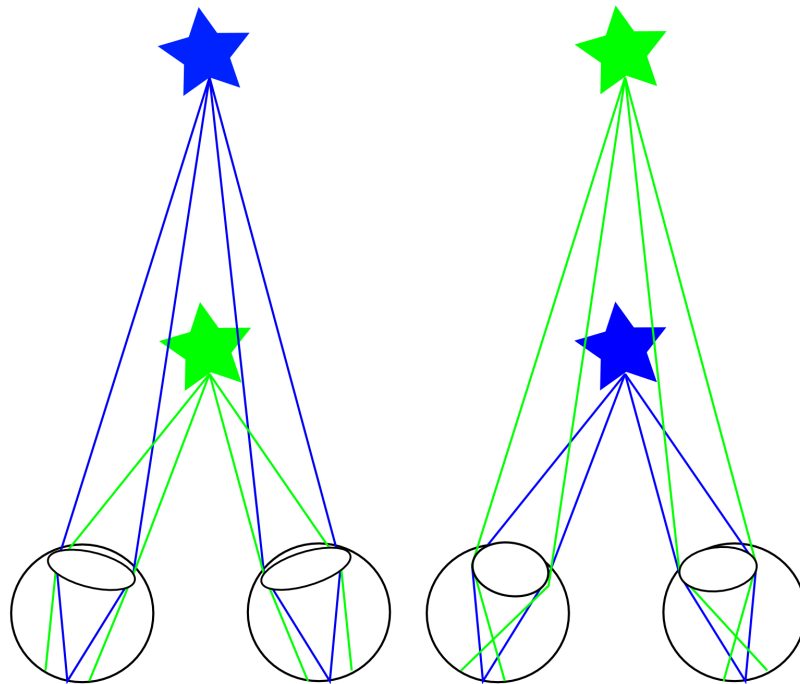


Figure 2.8: Inferring depth using the blur gradients from the accommodative effect. Rays from the object not in focus (green) do not form a sharp image on the retina, unlike rays from the object in focus (blue).

focal distance will be blurred, providing a cue as to their depth based on the degree to which they are blurred.

Due to the horizontal separation of the eyes, each eye needs to rotate inwards to form a single, clear image on the fovea of an object of interest (at the so called “fixation point”). This rotation effect is called vergence. Objects behind or in front of the fixation point will form images on the retina that are not on the fovea (Figure 2.9), and are thus blurred. This provides a depth cue as angular deviation from the fovea (and thus the blur magnitude) is an indicator of the distance between the fixation point and the unfocussed object.

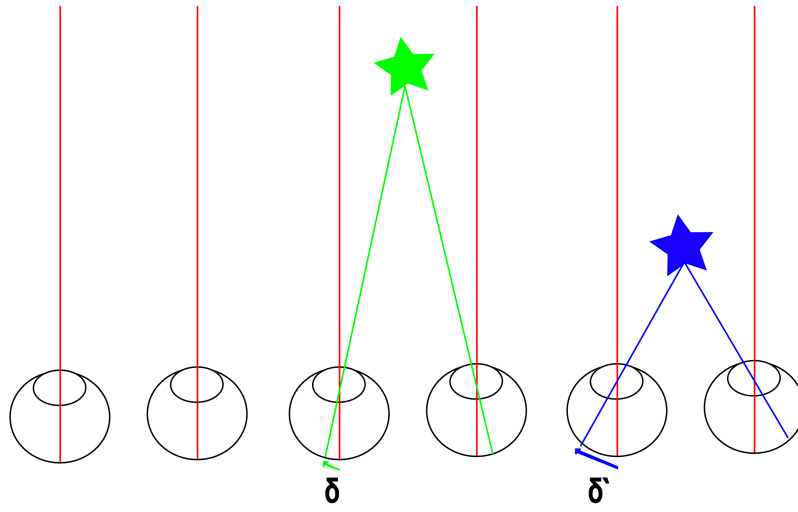


Figure 2.9: Where an image is formed on the retina relates to the disparity between the object location and the current fixation distance. When fixating at infinity (left), an object that is closer to the focal distance (centre) will have a smaller disparity on the retina than an object that is further (right), i.e. $\delta' > \delta$.

In stereoscopic displays, the accommodation and vergence depth cues do not match, as the eyes converge to a wide variety of depths, depending on the virtual depths of objects in the VE [43, 44, 45] (Figure 2.10). Only content rendered at a virtual depth of zero will ever eliminate this disparity, but this solution has two major issues: (1) if all content in a scene is rendered at zero depth, then the scene is 2D, which defeats the entire purpose of stereoscopic content, and (2) it is known that viewing content at zero virtual depth causes discomfort through other means [46].

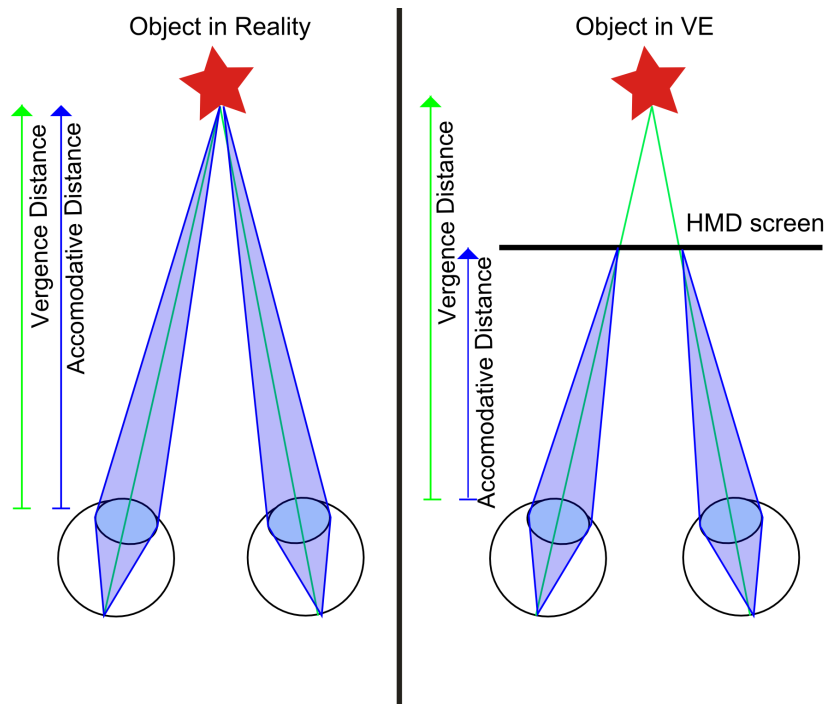


Figure 2.10: An illustration of the accommodation-vergence conflict, comparing visual situations between real life (left) and HMDs (right)

The result of this mismatch is a conflict in the expected depths of visual stimuli, which is known to be a cause of visual discomfort by causing “visual fatigue” [42, 47, 48, 49]. Furthermore, this conflict does not have a constant influence on visual discomfort: a greater rate of change in vergence demand results in greater visual discomfort when accommodative demand is fixed (as is the case for HMDs including the Oculus) [50]. This indicates that visual discomfort is not only caused by conflicting depth cues: the visual system having to re-adapt to new, differently conflicting cues is a cause of discomfort as well.

The accommodation-vergence mismatch has effects beyond causing conflicting depth stimuli: it also increases binocular fusion times and decreases fusion accuracy, decreasing the perceived quality of immersive 3D experiences even when visual discomfort is not considered [51]. It is con-

cluded that correct or consistent accommodation and vergence cues are critical to comfortable 3D viewing [42].

Long term adverse effects on the visual system are known to be a result of inconsistent accommodation and vergence cues [52], and even short term effects can be of concern, especially in tasks where accurate spatial reasoning is critical, such as when driving or operating machinery. For this reason, the US Air Force restricts pilots from flying real planes within 12-24 hours of exposure to a VR flight simulator [53].

2.4.3 Immersion and abnormal motions

The concept of presence refers to the illusion where a user believes they are physically located in the VE they are observing rather than in the real world. A VE with a greater presence illusion thus refers to a VE that induces a greater belief in a user's presence in the VE. VEs that cause stronger presence illusions are known to induce more extreme emotions and reactions in users, such as causing greater levels of fear [54]. As a result, some stimuli that would not be discomfoting when viewing content on traditional screens, such as high linear and rotational accelerations (or other such extraordinary motions) become discomfoting on HMDs [55].

2.4.4 Hardware considerations

As observed in Section 2.1, the fundamental design of commercially-available HMDs has not changed since 1960. This is unfortunate, as parts of this design are known to directly cause discomfort. For example, the chassis design of PI-VR HMDs occludes all peripheral vision to increase immersion, however this occlusion of all external reference points is known to result in greater discomfort [56].

The hardware specifications that are considered optimal for achieving user presence are beyond the level of current generation devices [57]. A general summary of the current “minimum specifications” for hardware to achieve optimal presence as are follows:

The Field of View (FOV) of the display must be $> 80^\circ$, as this is required to provide peripheral motion cues for motion, balance and situational awareness. The resolution of the display in the HMD needs to be, at minimum, $1080p$, as the small distance to a HMD screen from a user’s eyes results in individual pixels taking up a large angle in the visual field: with lower resolutions, the display appears “blocky” and the gaps between pixels are noticeable. The display FoV (DFoV) of the HMD should also match the geometric FoV (GFoV) of the displayed content, as if the DFoV does not exactly match the GFoV then retinal slip will occur when users move their heads, causing discomfort identical to that discussed in Section 2.4.1. The small separation of the HMD screen from a user’s eye also exacerbates the vection effect mentioned in Section 2.3.1, making consistent FOVs paramount to comfortable viewing [32].

The refresh rate of the display should be above $95Hz$ in order to avoid flickering or noticeable visual lag, and each pixel should not be lit for the entirety of a frame, (an effect referred to as “low persistence”), to avoid the visual field blurring during head motions. Low persistence displays are a comparatively recent technological development: the Oculus Rift DK2 is a low persistence display, but the DK1 is not. Likewise, the display should be “global” (as opposed to progressive): every pixel should update simultaneously to avoid blurring effects.

The optics within the HMD need to approximate the behaviour of the human perceptual system, which is difficult with a small number of lenses.

High quality positional and orientation tracking is required to allow the perceptual system to accept the VE as “real” (this arguably applies to both device and eye tracking).

The device should also have low “transport delay”²: below $20ms$ is currently accepted to be optimal. Historical studies into the transport delay on HMDs have concluded that there is a threshold value at which transport delay becomes noticeable and causes visual discomfort: previously proposed threshold values include $60ms$ and $48ms$ [58, 59]. None of this historical research was completed on systems capable of transport delays smaller than $48ms$, explaining why the threshold value has decreased over time as display technology improved.

In driving simulators, user actions can cause both visual and physical changes to their environment (physical referring to the simulator chassis moving, and visual referring to an update of visual information on a display). It was found that an increase in visual transport delay caused greater discomfort than equivalent increases in physical transport delay on these driving simulators [60].

Overall, the device must be optically well calibrated when compared to the behaviours of the human visual system: the human eye is sensitive to miscalibration, and presence requires the visual system to “believe” it is viewing real content. Failing to achieve optimal presence doesn’t just impact a user’s immersion in a VE; it also causes discomfort [57].

²The time between the user moving and the display updating, colloquially referred to as visual latency

2.5 Reducing Simulator Sickness

As previously discussed, repeated exposure to HMDs will reduce the severity and incidence of visual discomfort for the majority of users [10]. This is a process that can take one to two weeks in the case of VOR adjustment [11]. The time it takes to adjust to other causes of visual discomfort is less well studied, but known to be highly variable. In this section, some general strategies found in previous research for mitigating visual discomfort are discussed, and the strategy and implications of using DoF blur to reduce simulator sickness is investigated in detail.

2.5.1 Hardware solutions

Adjusting hardware factors to eliminate the issues discussed in Section 2.4.4 is an obvious solution for minimising visual discomfort. Relying on such advances in technology is not an optimal solution: such improvements in technology come at significant cost. For example, displays that are both high resolution and low persistence are expensive, and thus currently ill-suited to consumer HMDs. Likewise, 1080p screens that refresh a stereo scene at 95Hz require powerful GPUs that are too expensive for widespread consumer ownership at this time.

Some hardware factors have improved in a manner that does not make their cost prohibitive: the Oculus used for this research has a transport delay between 2-18ms for head motions³, and 32-48ms for interactions with the VE (firing a weapon, flipping a switch, etc.) [61]. This latency is significantly lower than possible with previous HMDs (such as those discussed in subsection 2.4.4). As previously mentioned, 20ms is the currently ac-

³This variability in latency is caused by the progressive rendering on the LCD in the Oculus, where the top line of pixels are updated before lower lines are. This is the behaviour global displays were designed to eliminate.

cepted maximum transport delay for comfortable and immersive viewing: latencies of this magnitude are generally imperceptible [61].

It has been suggested that modifications to the fundamental design of the hardware used for viewing VEs are required for decreasing visual discomfort [56]. Some of these new designs focus on specifically solving the accommodation-vergence conflict to reduce visual discomfort. Multifocal displays [62, 63], alternative lens systems [64] and multi-lens systems [2, 65] have all been proposed as methods for effectively reducing simulator sickness by providing close-to-correct accommodation cues for multiple depths. However, their complex construction prohibits easy application to consumer level devices such as the Oculus (at this time).

2.5.2 Software solutions

It has been found that VEs in binocular HMDS cause less discomfort when the scene is rendered in a monoscopic manner, compared to stereoscopic rendering [66]. As monoscopic rendering provides no binocular depth cues, it was concluded that removing these cues can reduce visual discomfort. The presence of binocular depth cues thus has an effect on visual discomfort, which is supported by research showing that manipulating depth cues through methods such as defocus blurring increases viewing comfort [67].

The developers of VEs can also take steps to mitigate visual discomfort, by reducing the rate at which a user's vergence must change [50], or by only displaying content in a stereo "zone of comfort", such as that proposed in [68]. These techniques work well for film applications where the director has significant control over the entire visual scene, but have limited application in user-dependent real-time activities such as gaming [69].

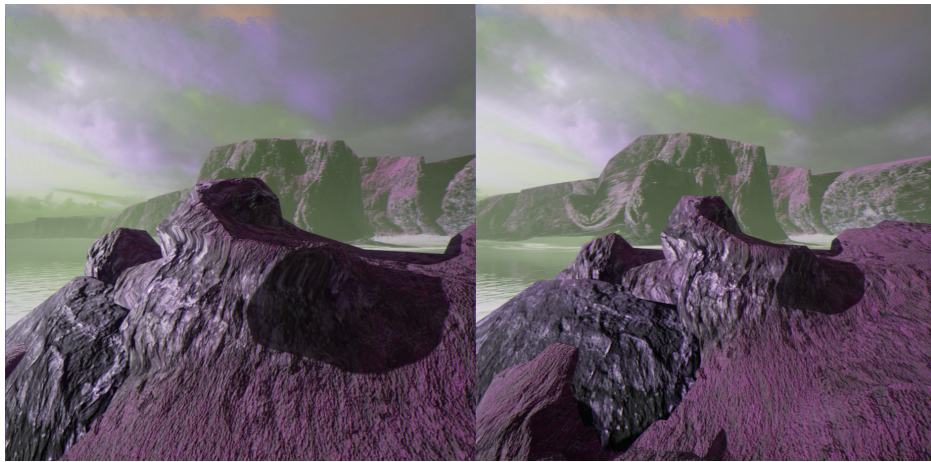


Figure 2.11: An orientation-independent (fixed) background reduces visual discomfort. Notice how the camera has rotated around the foreground rocks, but the same view of the sky and clouds is presented from both the left and right perspectives.

Without overly restricting the range of content that can be rendered as the previous two solutions would, alternative rendering methods can be considered for manipulating user reactions to VEs. One such method is to fix the orientation of the VE's skybox with respect to the user so that an identical backdrop is viewed no matter the user's orientation (Figure 2.11). This unusual rendering technique reduces visual discomfort, as the sensory mismatch where the user perceives self-motion is mitigated when this portion of the visual field undergoes no relative motion [70, 71]. This allows the user's brain to form an interpretation where the vestibular and visual stimuli are consistent: the user is perceived to be stationary along with the background environment and the foreground environment is perceived to be moving.

2.5.3 Using DoF blur to reduce simulator sickness

The accommodation and vergence depth cues are known to conflict in normal stereoscopic HMD viewing, and it is concluded that addressing this conflict may decrease visual discomfort [48, 47]. One proposed software implementation for reducing the impact of the accommodation-vergence conflict is to apply a Depth of Field (DoF) blur to the VE based on where the user is known to be looking, mimicking the blurring of out-of-focus content in reality. This is known as “Gaze-contingent DoF” and in specific VR viewing applications, such as when viewing stereoscopic content on LCD screens through a haploscope, has been shown to reduce visual discomfort [72].

Adding blur gradients to simulate DoF and peripheral blur has been shown to improve the quality and realism of game play in desktop LCD displays [73]. DoF blur also can reduce rivalry from monocular regions in stereoscopic images, improving both the viewing comfort and the quality of VEs [74]. DoF blurring masks the disparity between differing accommodation and vergence cues, as precise estimation of the disparity between the cues relies on high frequency spatial information [47, 75]. Other studies have shown DoF blur effects ease fusion of stereoscopic content, increasing viewing comfort [67].

Focal cues including artificial blur directly contribute to the quality of a 3D experience through the reduction of visual fatigue [76], and correcting these cues is considered one of the most important factors for improving viewing comfort [77]. Correct focusing when viewing content on a stereoscopic display was shown to reduce visual fatigue and discomfort by lessening the strain caused by the accommodation-vergence conflict [48].

It is thus reasonable to conclude that DoF blurring can be used to im-

prove viewer comfort in stereoscopic displays. In addition, implementing gaze-contingent DOF allows users to perceive images as more realistic, with an increased sense of depth [78]. Such blur effects are known to drive “allocation of user focus” [79]: strong biases are shown towards sharp regions in images with DoF blurring [80]. Given that DoF blurring does not drive the accommodative response of the visual system [81, 82], this research proposes that gaze dependent DoF blurring can be used to modify visual behaviours of users on HMDs by reducing both the area of the screen that is in focus at any given time and the range of virtual depths a user must focus on.

A survey completed at the end of 2014 found that no prior perceptual studies have been carried out to report the impact of using DoF blur to reduce accommodation-vergence conflict in the specific case of stereoscopic HMDs [83]. This research thus proposes to investigate the hypothesis that DoF blurring effects will improve user comfort on PI-VR HMDs.

Chapter 3

Estimated Gaze Dependent DoF

The state of hardware used in current generation HMDs falls short of the thresholds introduced in Section 2.1 for comfortable immersion and viewing. Consequently, as previously discussed, the majority of users of HMDs experience visual discomfort, even after very small exposure times. Prior research into alleviating or eliminating this visual discomfort through software-only solutions has not had sufficient impact in resolving this issue. In order to fulfil the thesis objectives, a software defined solution that effectively reduces this visual discomfort is required. A depth of field (DoF) system was proposed in Section 2.5.3 to fulfil this requirement. This research proposes that such a DoF system can be implemented using an estimate of a user's gaze to find the focal depth, instead of requiring eyetracking (resulting in the solution being named "Estimated Gaze Dependent Depth of Field" (EGD-DoF)).

In this research, an effective EGD-DoF algorithm is presented. Ideally, this algorithm will:

- Be fast enough that a reasonable implementation is renderable in real-time on consumer level hardware.
- Employ the subjective measure of "a user can or will not notice any adverse impact of the DoF system". In other words, the DoF must not degrade the quality of a VE once applied.
- Be designed in such a manner that it can be applied to a variety of 3D applications without requiring significant modification.
- Be implementable and testable on scenes of sufficient quality (i.e.

high resolution textures and high-poly models as would be expected in a modern game) that a user may experience place illusion. Without this place illusion, immersion related factors that induce visual discomfort cannot be investigated.

In this chapter, the algorithms that comprise the core of the EGD-DoF system are introduced and discussed. An implementation of these algorithms is later discussed in Chapter 4.

3.1 Overview of the EGD-DoF Algorithm

The EGD-DoF algorithm consists of two major parts: A gaze estimator and a DoF blurring algorithm (Figure 3.1). The gaze estimator uses 3D information that can be gained from the 3D scene and motion data provided by the HMD to estimate a user's gaze. Using this estimate, the optimal focal depth of the scene is calculated and passed to the DoF algorithm. This algorithm then blurs the scene before it is rendered to the HMD.

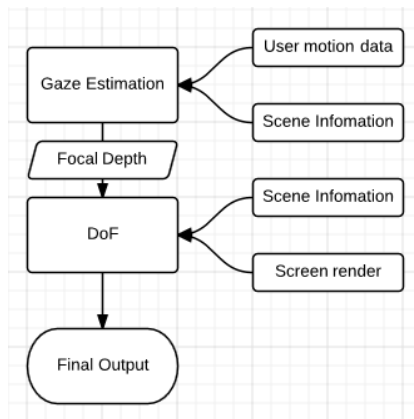


Figure 3.1: Overview of the EGD-DoF system.

3.2 Gaze estimation

3.2.1 Estimating the user's gaze direction

In typical gaming conditions, approximately 86% of a user's fixation time, and 82% of their total viewing time is spent looking the centre of the screen [84, 80]. In Section 2.5.3 a summary of prior research showed that DoF blur effects drive allocation of user focus. Hence, if a DoF blur was applied such that the centre of the screen¹ was in focus, the combined effects of the blur and centre of screen fixation bias should result in the user fixating at the centre of the screen for the majority of their HMD exposure.

We thus assume that the user is always focussing at the centre of the screen, in the absence of stimulus that would shift their focus (such as a crosshair). This assumption allows for the construction of a DoF algorithm that is gaze dependent without requiring eye tracking. We are not aware of any consumer level current generation HMDs that have inbuilt eye tracking. This means that such gaze assumptions are the only way to implement gaze dependent DoF in a HMD without violating the thesis objective that states a software solution should be constructed without requiring external hardware devices.

3.2.2 Finding the focal depth

Since it is assumed that the user will always be looking at the centre of the screen, the ideal focal depth of the scene d_f is the depth of the object under the central pixels of the screen. Finding the depth of the object underneath the central pixel is implementation dependent, and thus is discussed in Chapter 4. We will thus assume d_f is known for the gaze estimation algorithms.

¹In a binocular screen this "centre" is defined as the centre of the virtual viewport formed by combining the two images.

The Oculus used in this research has a rather low resolution screen (640×800 per eye, well below the $1080p$ mentioned in Section 2.4.4). The angular distance between pixels (as seen by the user of a HMD) is much larger than plausible angular differences between objects in the VE. As a result, it is difficult to determine whether the object exactly in the centre of the screen (in screen-space) is the actual object being focussed on, especially if foreground images have a small gap between them, as shown in Figure 3.2.

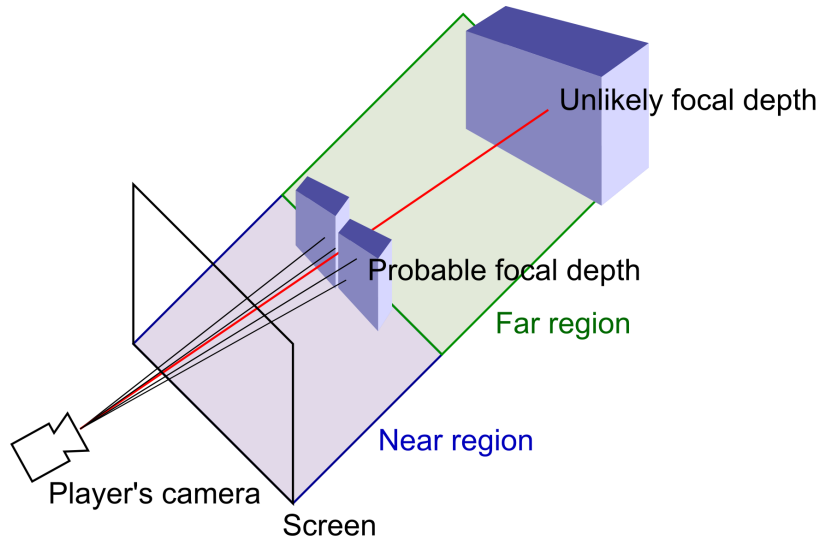


Figure 3.2: In order to avoid focusing at unlikely depths (red), multiple rays (black) are cast to find d_{new} .

In the case that a viewing situation similar to Figure 3.2 occurs, the gaze estimation algorithm attempts to find the optimal focal depth, such that the focal depth always matches the user's expectation. In order to do this, once d_f is known (the depth at which an object was intersected along \mathbf{v}), eight new rays are cast into the scene and the depths that these eight

rays intersect objects are reported (Algorithm 3.1).

Algorithm 3.1 Construct rays for finding the optimal depth for user focus

Assume the scene is rendered from the viewport of a camera, c .

Let \mathbf{p}_c be the scene-space location of c .

If a ray is cast from the player's eye through the centre of the screen:

Let \mathbf{p}_f be the point in the scene that this ray first intersects an object, at the depth d_f .

Let $\mathbf{D}_{1..8}$ be the vectors along which new rays will be cast

$$\mathbf{v} = \mathbf{p}_f - \mathbf{p}_c$$

$$r_a = \frac{d_f}{80}$$

for $i = 0; i < 8; i++$ **do**

$$\theta = (i \times 45)^\circ$$

$$x = r_a \times \cos(\theta)$$

$$y = r_a \times \sin(\theta)$$

$$\mathbf{r} = \langle x, y, 0 \rangle$$

$$\mathbf{D}_i = \mathbf{v} + \mathbf{r}$$

Let d_i be the depth of the object first intersected by the ray \mathbf{D}_i .

end for

Each of these rays are at small angular deviations to \mathbf{v} . If the central ray is reporting a suboptimal depth, then these rays should intersect objects at different depths and thus the optimal depth can be found. To do this, the nine known object intersection depths $d_f, d_{1..8}$ are processed using Algorithm 3.2.

Algorithm 3.2 Calculate d_f

Let d be the set of nine object intersection depths

$$\mu = \frac{1}{9} \sum_{i=1}^9 d_i = \text{mean of } d$$

$$\sigma = \sqrt{\frac{1}{9} \sum_{i=1}^9 (d_i - \mu)^2} = \text{standard deviation of } d$$

while $\exists d_i$ such that $|d_i - \mu| > 2\sigma$ **do**

 Find $d_m = d_i$ such that $|d_i - \mu|$ is maximised

 When multiple d_m candidates are found, select d_m to maximise $d_i - \mu$

 Remove d_m from d

 Recalculate μ and σ

end while

d_{new} = the mean of the remaining rays in d

The result of this algorithm is a depth d_{new} which is the optimal focal depth for the scene, given the current estimate of the user's gaze.

3.2.3 Updating the focal depth

As the user's neck moves, a corresponding movement of the player camera will occur in the VE, changing the depth of the object under the centre of the screen. Immediately changing to a new focal depth would be jarring for users, as the human eye takes up to 500ms to refocus (assuming a typical adult's eyes, refocussing over a large distance [85]). The time taken to refocus across a given distance difference depends on age, condition of the eye and light levels. For simplicity, this research assumes all users will take a static 500ms to refocus from an infinite distance to a close distance ($\approx 1m$), using the value calculated in [85]. Prior research involving gaze-driven rendering effects has used a value of 370ms and received user complaints about the focus time being too short [80], so we believe this value is justified. As a result an interpolation between the current and previous focal depths must occur whenever a user moves their head.

In order to complete this interpolation, Equation 3.1 is defined. Using a linear interpolation factor $l(t)$ in the range $[0, 1]$, where $l(0) = 0$ and $l(t_{max}) = 1$, d_{new} as the depth calculated as the current optimal focal depth in the previous subsection, and $d_{prevrender}$ as the previous focal depth, the new focal depth d_{render} is calculated:

$$d_{render} = (l(t) * d_{new}) + ((1 - l(t)) \times d_{prevrender}) \quad (3.1)$$

Using this interpolation, the focal depth of the scene will smoothly interpolate between focal depths as the user moves their head. By modifying the behaviour of $l(t)$ with respect to t , the rate of interpolation can be controlled. As previously mentioned, this research assumes a refocussing period of $500ms$ for the focus to move from infinite distance to $\approx 1m$. Therefore the exact refocussing velocity will depend on the implementation specific depth that determines infinite distance (typically the virtual depth of the skybox).

3.2.4 Gaze estimation summary

A summary of the algorithms that comprise the “Gaze Estimation” system is shown in Figure 3.3. This entire figure is equivalent to the Gaze Estimation node shown in Figure 3.1.

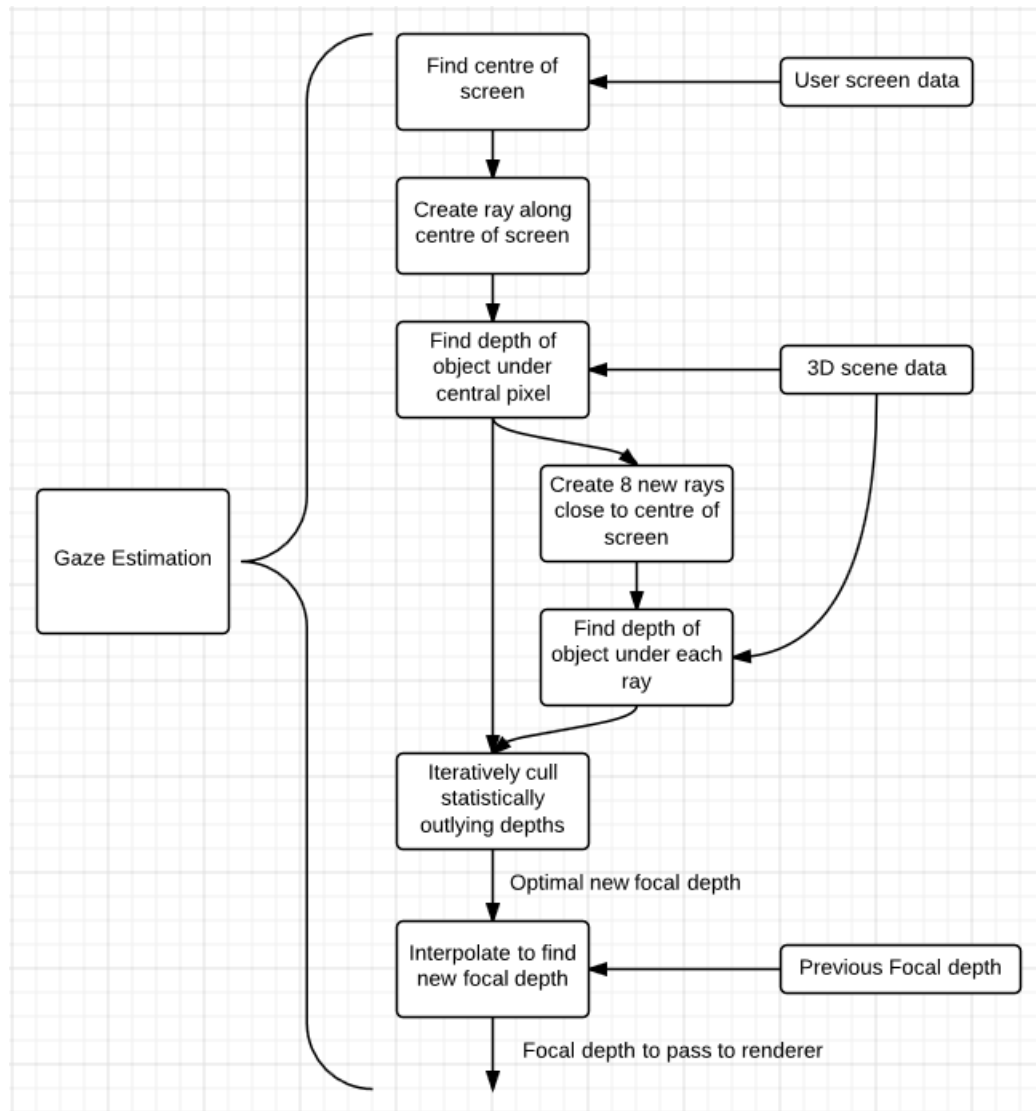


Figure 3.3: Overview of the algorithm that finds the focal depth based on an estimate of the user gaze direction.

3.3 Real Time Depth of Field

In this section, multiple algorithms for achieving DoF blur effects are introduced. Due to the constraints introduced at the beginning of this chapter, only DoF algorithms that can be implemented as a post-process tech-

nique are considered: other methods including wave-propagation and distributed ray tracing require fundamental changes to how an image is rendered and thus lack the required flexibility and applicability.

DoF is a phenomenon that arises due to the nature of lens optics: it is not possible to have light rays from an infinite range of distances collate at a single point once they have passed through a focussing lens (such as those present in the human visual system). Any lens system will have a focal point at which light rays from a point source will collate, and all light rays from differing distances will instead project to a region. This region is referred to as the “Circle of Confusion” (CoC) [86]. The diameter of a CoC increases with lens size and distance from the focal plane of a lens. A non-zero CoC implies an object has its light rays projecting across a non-point region: this causes a blur effect. In practice, CoCs that are very small are unnoticeable, especially if they are smaller than the resolution of film (in photography) or the size of a pixel (in computer graphics). As a result, a visual field viewed through a lens can be said to consist of three regions: a blurry foreground, a sharp (in-focus) midground, and a blurry background.

In computer graphics, an idealised pinhole camera with a zero size lens is typically used. No natural DoF will occur as the resulting CoC diameter for such a camera is 0. As a result, the depth of field effect must be algorithmically approximated as a post-processing step [87]. Modern, real-time DoF implementations are typically achieved on the GPU, such as in [72, 86].

Some of the more popular post-process algorithms for achieving DoF blur are Poisson disc sampling, Gaussian blur and bokeh² blur [88]. Poisson disc sampling, such as that implemented by [89] is effective at blurring

²From the Japanese word *boke*, meaning blur. Defined as “the way the lens renders out-of-focus points of light”.

background objects but results in poor blurring of foreground objects [90]. This research thus focusses on the Gaussian and bokeh DoF post-process techniques as candidates for implementation.

3.4 Selecting the Blur Algorithm

Gaussian DoF uses a fast algorithm where all points outside the focal plane are multiplied by a simple Gaussian kernel to give a blurred image. By adjusting the width of the Gaussian kernel, the degree to which an area is blurred can be changed: a higher width kernel will blur more than a smaller width kernel. Since the entire blurred portion of a scene is being multiplied by a Gaussian kernel, the output of Gaussian DoF is comparatively smooth and the balance between light and dark areas in the foreground and backgrounds will be preserved.

Bokeh DoF emulates the effect of lens aberrations and differing lens aperture on rendered scenes, where the circle of confusion for each point in the scene is not necessarily a circle: each point becomes an image of the shape of the lens aperture. Lights also become significantly more pronounced using bokeh blur. The effect of bokeh blurring is shown in Figure 3.4: the small heart shaped lights take on the ovular shape of the camera lens and dominate the darker areas of the image when blurred. As the human eye is also a lens system, with the correct parameters bokeh can better emulate the DoF effect of the human visual system, due to the circular shape of the iris, compared to Gaussian DoF.

An example of the difference in the output of each algorithm is shown in Figure 3.5. This figure shows a scene rendered in Unreal Engine 4 (UE4) that has had each of bokeh and Gaussian DoF applied as a post process. The artificial circular bokeh aperture is particularly noticeable in the foreground of the bokeh example, compared to the consistent effect of the



Figure 3.4: A photo without bokeh (left) compared to the same subject with bokeh blur (right). Image by Arunus, from en.wikipedia.org

Gaussian blur.

As bokeh DoF produces images that can be more accurate to the human eye, this research will use bokeh as the basis for its DoF implementation.

Achieving a bokeh blur effect in a rendered scene can be accomplished in two ways: the first is to model a camera with lens aberrations when rendering the scene, the second is to approximate the effect such a lens would have on the scene as a post-process. As previously stated, computer graphics applications often use an idealised pinhole camera, so the first option is not feasible to implement. The bokeh effect is thus achieved through a post-process algorithm. Algorithm 3.3 shows a basic example of

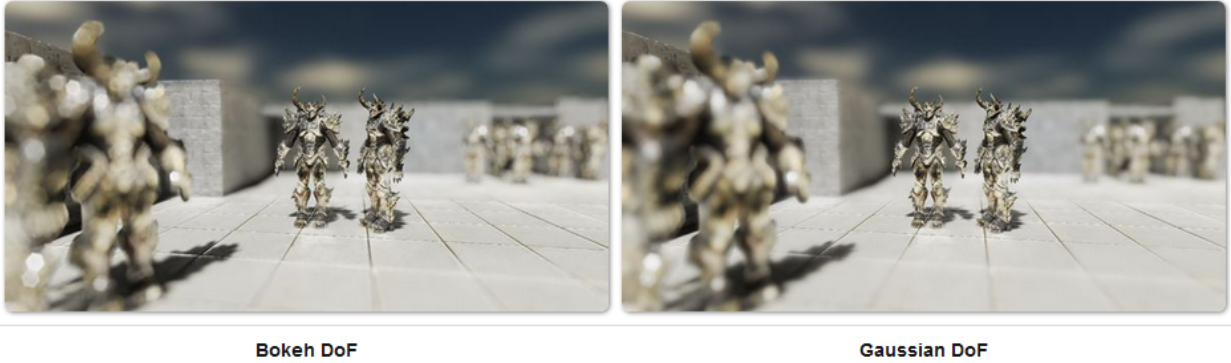


Figure 3.5: Bokeh (left) and Gaussian (right) DoF have significantly different outputs in UE4. Image from [91]

such a post-process implementation, assuming a circular kernel like the iris in the human visual system.

Algorithm 3.3 Estimating the bokeh effect using postprocessing

Given $p_{0..n}$, the set of all pixels
for Each pixel p_i **do**
 Calculate d_i , the depth of the object under pixel
 Calculate the radius r_i for the corresponding CoC
 Generate the corresponding multiplication kernel M_i
end for
Create an output image Q consisting of pixels $q_{0..n}$
for Each pixel p_i **do**
 Multiply p_i by the corresponding multiplication kernel to get N_i
 Add N_i to Q , centred at q_i
end for
Return Q

This algorithm essentially creates a texture containing a circle of radius r_i , for each pixel in the image. These textures are multiplied by the corresponding pixel colour value and summed to create the output image (a process that can be considered equivalent to a convolution of the input image against a circular kernel of changing radius). This naive implementation can suffer from a few issues: (1) it is very slow, requiring multiple calculations for every pixel displayed, and (2) as the output Q is the result of a sum across all the blurred pixels in the scene, it is possible that the blur from objects further away from the camera might occlude objects closer to the camera (a so-called “bleeding effect”). Chapter 4 details how these issues can be addressed by the chosen implementation of bokeh DoF.

The presented bokeh DoF algorithm can be summarised by Figure 3.6 and can be considered equivalent to the DoF node in Figure 3.1.

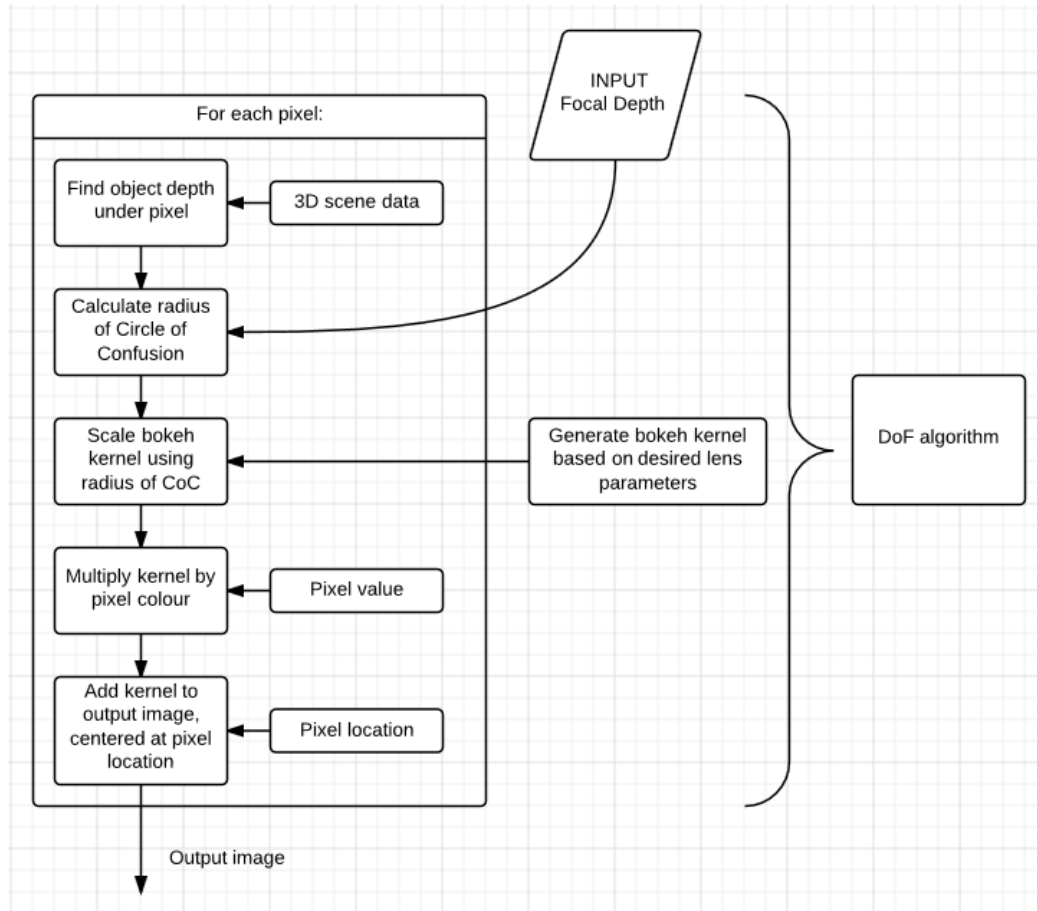


Figure 3.6: Overview of a basic implementation of bokeh DoF.

Chapter 4

Software Implementation

In the previous chapter the framework of a EGD-DoF system was laid out, and constraints that define whether such a system would be effective or not were introduced. Bokeh DoF blurring was noted to have the capability to be more realistic to human vision than other blurring methods, and as a result was selected as the initial DoF blurring technique. This chapter details an implementation of the EGD-DoF algorithm in a commercial game engine (Unreal Engine version 4, or UE4) and introduces a game-based environment that is used for user interactions with scenes where the EGD-DoF system can be evaluated.

4.1 Unreal Engine v4

UE4 was selected to be the engine in which the EGD-DoF algorithm was implemented for the following reasons:

1. UE4 is primarily a game engine. This means it is optimised for creating game environments, which is ideal for evaluating the EGD-DoF system: the ideal participants for an evaluation of this system are students, who tend to be familiar with game environments.
2. UE4 has high quality implementations of both bokeh and Gaussian DoF as post-processes. This allows a high quality DoF implementation to be created, and if bokeh DoF is too slow to achieve the constraints mentioned in Chapter 4, then Gaussian DoF can be implemented without significant modification.

3. UE4's blueprint editor allows for fast and intuitive construction of algorithms such as EGD-DoF without requiring the learning of a new programming language or API.
4. The variety of high quality free content available in the Unreal Marketplace¹ allows the EGD-DoF system to be tested on a variety of environments without having to manually build such environments as part of this research.
5. UE4 provides live debugging: the time it takes to compile a blueprint and test it is typically under 30s, in contrast to the significantly longer compile times required in other engines.

4.1.1 Blueprints and the UE4 editor

UE4 offers multiple editing interfaces for the design and implementation of projects: content can be built using C++ code, or it can be built using "blueprints". The UE4 blueprint editor is a visual front-end that allows for intuitive controlling of both user-defined and native UE4 code through drag and drop elements. These blueprints are able to control a significant portion of the implementation of a project: assets can have their visual and interactive behaviours defined, a player's Heads Up Display and avatar can be constructed and manipulated using blueprints and post-process effects such as DoF can be controlled. There are limitations: for example, complex mathematics and methods that require non-basic iteration are not easily or intuitively implementable in blueprints.

The majority of the EGD-DoF system developed in this research was built using the blueprint editor. In order to understand the blueprints referenced in this chapter, a few elements require explanation. Using Figure 4.1 as an example, with each element labelled (1) through (9):

¹An online store where content producers can sell virtual goods ranging from simple models to complicated landscapes, all of which are compatible with UE4.

- Each small box (1) (not the large grey ones) is called a “node” and refers to an individual function. These take inputs from the left, and return outputs on the right. The behaviour of a node can be defined as a blueprint, or as a C++ function.
- The interior of each large grey box (2) refers to the components of a node. These can have input parameters, which are brought into a function in a box titled “input” (by convention at the left of the node), and return outputs which are likewise represented by a box titled “output” (by convention) at the right of a the node.
- Different colours represent different types of information. This may be a line (3) showing information being passed between nodes, a *setter* node that sets a variable’s value (4) or a function input/output (5):
 - White lines (3) show execution order. Once every section of code represented by a node finishes executing, the subsequent node (linked via these white lines) will begin execution.
 - Other colours show different variable types: green refers to floats, booleans are red, rotations are light blue, vectors are orange, integers are teal, object names are purple and object references are blue.
- Conditionals are grey function boxes (6) that split the execution line into multiple possible nodes.
- There is no parallel execution within a blueprint: sequences can be defined but each component will compute one at a time, with some exceptions (see red nodes, below).
- Different coloured nodes represent the different “classes” of node.
 - Red nodes (7) deal with the beginning of an execution event (such as when the user pushes a button). These are the only

nodes that can allow simultaneous execution, typically when multiple events trigger at once (i.e. when the user pushes the “jump” and “forwards” keys at the same time, both the jump and move forwards actions should be simultaneously executed)

- Blue nodes are either functions or events. They can be differentiated by the icon next to the name of the node. An example of each is shown in Figure 4.1.
- Grey nodes are either macros (sequences of nodes collapsed for visual brevity into a single node) or a flow control node such as conditionals (left of 3).
- Green nodes (8) are generic functions.
- Nodes with a dot in the centre (9) convert between data types

For the remainder of this chapter, names introduced in *italics* are the names of the variables used in UE4 blueprints.

It should be noted that some function parameters have default values which are not modified in this research: typically these are often not visible, as many UE4 functions take large numbers of parameters. Important parameters that have been set to use their default input have had their visibility flag toggled so that the parameter and its value can be seen.

4.2 Overview of the EGD-DoF implementation

An overview of the entire EGD-DoF system, showing each of the blueprints that will be introduced in this section and how they are linked can be found in Appendix A.

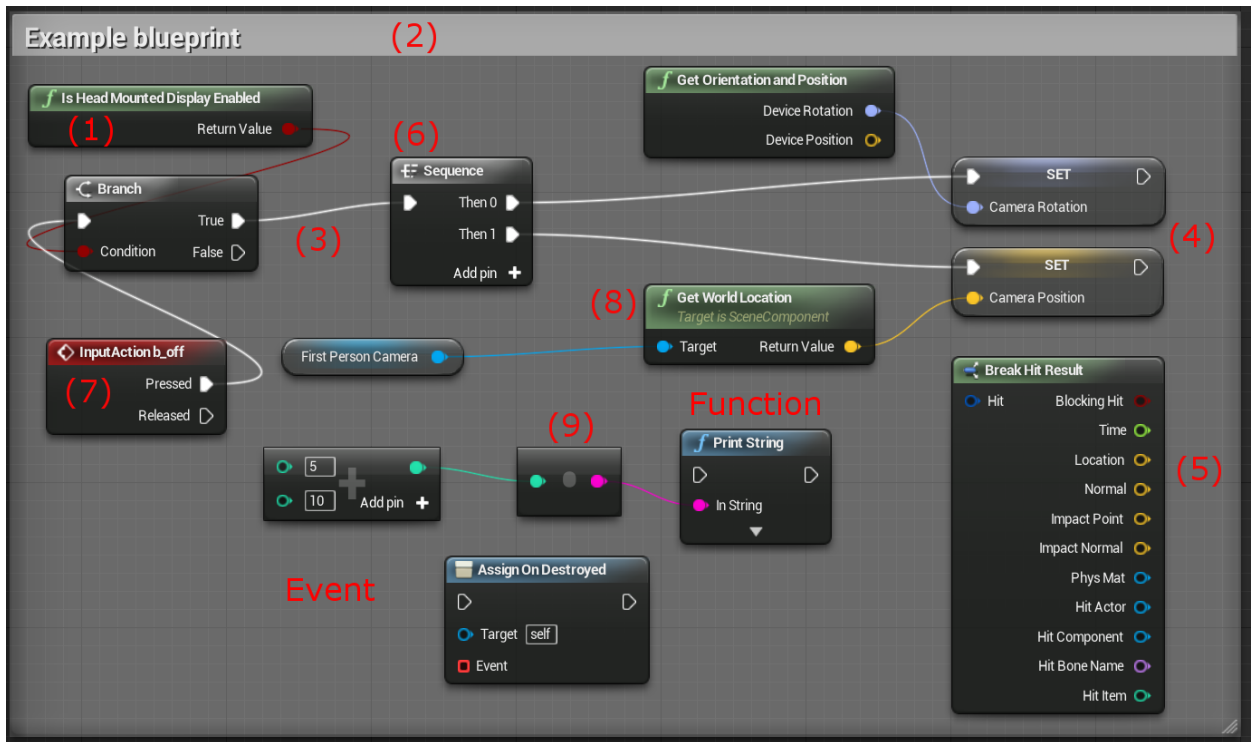


Figure 4.1: An explanatory UE4 blueprint, highlighting each of the different types of components used in this research.

4.3 Implementing Gaze Estimation

In order to implement gaze estimation, three variables must be known: (1) the position of the camera that forms the viewport that the user will look through must have a known transformation relative to what is being viewed (the scene), (2) the orientation of said camera relative to the scene must be known, and (3) the transformation that describes what the player sees on the HMD screen compared to what the camera “sees” in the scene is required. This allows the system to cast rays at the central pixel (what the player sees) and find an object depth (what the camera sees).

In this implementation the camera that forms the user’s viewport is

permanently mounted to an avatar. In order to solve for the above transforms, it is first necessary to mention that the control scheme implemented for interaction with the scenes allows for the user to independently control the orientation of the camera (using physical motions of the HMD) without affecting the orientation of this avatar. More detail on the avatar and this control scheme is presented in Section 4.5.

To find (1), we can simply query the location of the camera in scene-space. To solve for (2), we find the rotation of the avatar that the camera is attached to, and then add that to the rotation caused by the user's neck motions. In this implementation, this addition does not need to occur if the HMD is disabled or not tracking neck motions. (3) is automatically handled by the UE4 - Oculus integration extension, where the software profile of the Oculus is queried to construct the pair of virtual cameras (one for each eye). As such, we can treat the centre of the camera as the approximate centre of both viewports on the HMD.

The blueprint that controls this behaviour is shown in Figure 4.2. Green *getter* nodes query the scene-space rotation and orientation of the camera, as well as the orientation and position of the HMD. If the HMD is enabled and responding, these orientations are added together and set as the variable that represents the rotation of the camera *Camera Rotation*, otherwise just the orientation of the camera is used. Regardless of whether the HMD is working or not, the *Camera Position* variable is set to the current camera position. These variables are used instead of simply querying the camera for its position and orientation as required because, as previously mentioned, execution in blueprints is not performed in parallel. This means that when we cast multiple rays later in this sequence, if the player moves significantly, then the rays will be cast with potentially unanticipated and/or undefined behaviour.

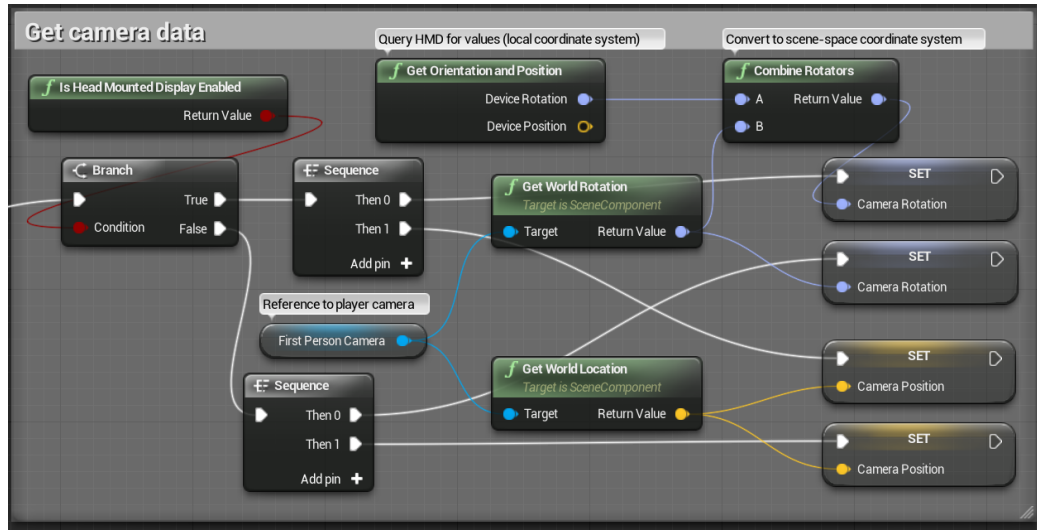


Figure 4.2: Finding the position and rotation of the player’s camera.

In this implementation, a coordinate system separate to both scene and screen-space is defined, which can be called camera-space. In camera-space, a ray cast from a user’s eyes through the centre of the screen is defined to be along the vector $\langle 1, 0, 0 \rangle$. Another aspect of the control scheme covered in Section 4.5 is relevant here: the torso of the avatar that the camera is attached to has a locked pitch and roll, and its yaw is controlled by the user’s mouse (the camera roll, pitch and yaw are controlled by motions of the HMD as would be expected). As a result, given the mounting of the camera to the avatar is done such that $\langle 1, 0, 0 \rangle$ in camera space will point in the direction the avatar’s torso “faces”, the transformation between camera-space and scene-space (i.e. (3) from earlier in this section) is simply the *Camera Rotation* variable mentioned earlier.

This transform is relevant when casting rays into the scene to find the object depth under the central pixels. To cast a ray into a scene in UE4, a point in scene-space is required defining where the ray should be cast from, and where it should be cast to. To find this *Trace End* value, the “Set Trace End” blueprint shown in Figure 4.3 is constructed. The x-vector of

the camera is found and multiplied by a very large integer (the reasoning behind this specific number is discussed in Section 4.6. For now, it suffices that 1 “unreal unit” as they are seen here is equivalent to 1 cm). When this is added to the position of the camera, a point is defined sufficiently far away (5 km) from the camera (along $\langle 1, 0, 0 \rangle$ in camera-space) that a ray that reaches this length probably isn’t going to hit anything.

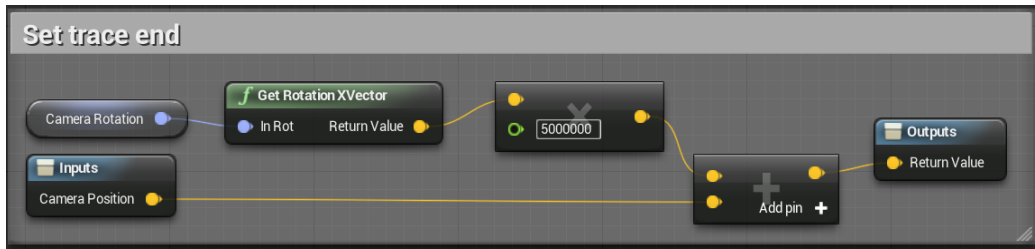


Figure 4.3: Setting the target point for rays cast into the scene.

This *Trace End* point can then be passed into the “Trace From Player” blueprint in Figure 4.4. Here, a node titled “Live Trace by Channel” takes the *Camera Position* and *Trace End* variables and sets it to be the *Start* and *End* respectively of a ray that is traced along until either an object is intersected, in which case a hit result and True Boolean are returned, or the trace reached the *End* point, in which case a hit result and False Boolean are returned. These hit results are basically meaningless except for packaging lots of sub-variables together, so the “Break Hit Result” node separates them into, among others, a point titled *Impact Point* corresponding to where in scene-space the intersection occurred. By subtracting the *Camera Position* from this point and taking the length of the resulting vector, we find *Trace Depth* which corresponds to d_f in Section 3.2.2.

With this initial depth value d_f / *Trace Depth*, the eight rays with small angular deviations to the first can be cast. This is implemented using the “Cast for Correct Depth” blueprint shown in Figure 4.5. Taking a *Focal Depth* parameter, which is the *Trace Depth* value from before, but renamed, and a *Trace Start* variable which is the *Camera Position* variable renamed,

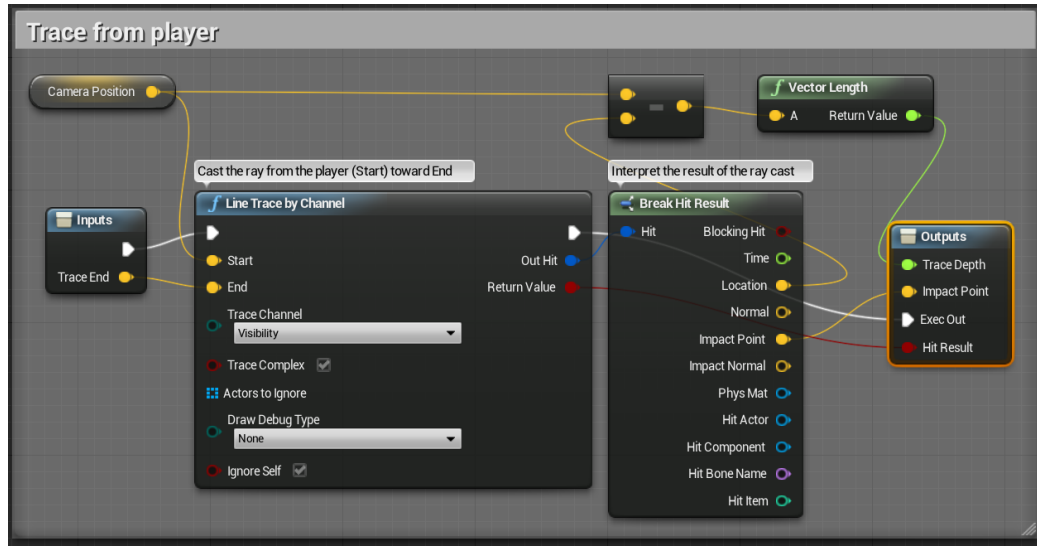


Figure 4.4: Casting a ray into the scene.

an array is initialised. A `for` loop then loops over the integer values $i = 0$ through $i = 7$, setting a variable called *Delta Rotator* to be $i \times 45^\circ$. Another *Trace End* variable is then constructed, using a point returned by the “Set Trace End” blueprint shown in Figure 4.3 added to a small vector $\langle 0, \frac{|d_f|}{80}, 0 \rangle$ which is rotated around the x-axis by *Delta Rotator*. A ray is then cast from the camera position to this new point using the “Trace From Player” blueprint from Figure 4.4, and the resulting ray depth is added to an array. Finally, a node called “Select correct focal depth” is called. This node calls behaviour that isn’t defined by blueprints. As previously mentioned, implementing mathematics in blueprints is unintuitive, so this node simply calls some C++ code identical to that in the `while` loop of Algorithm 3.2.

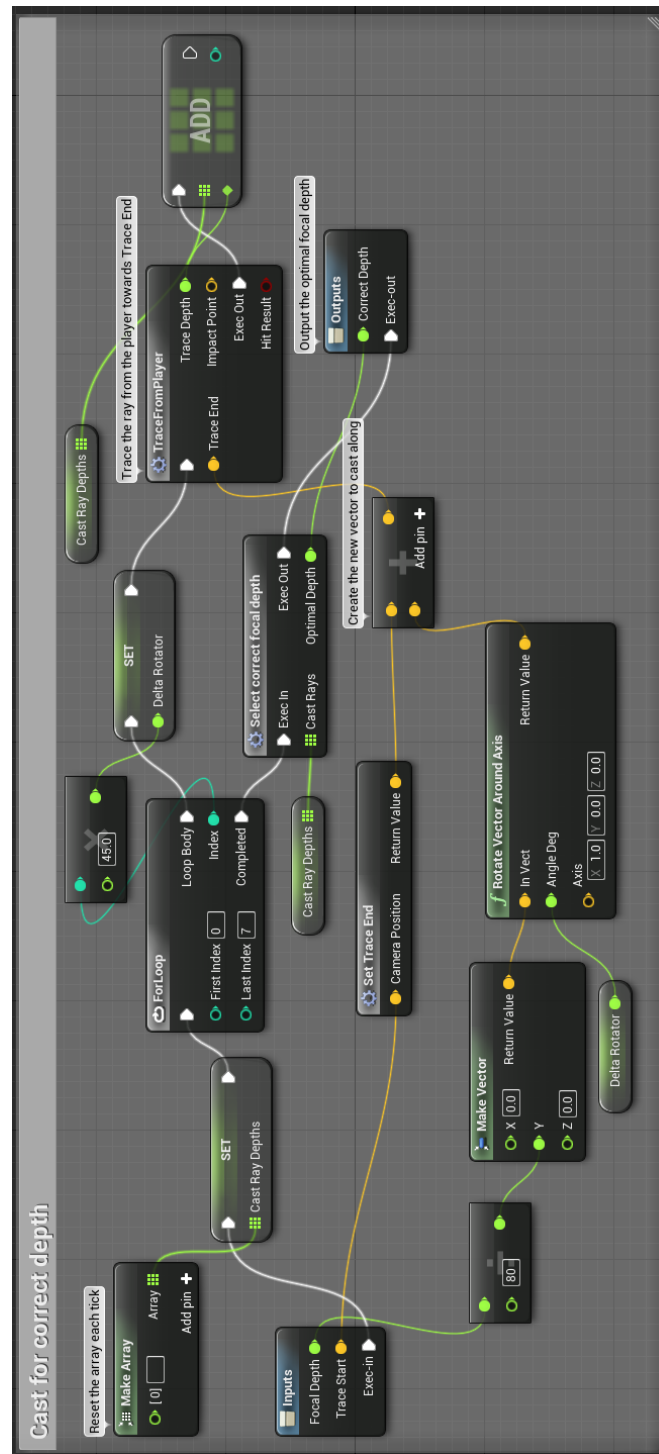


Figure 4.5: Casting eight rays at small angular deviations into the scene.

The end result of the “Cast for Correct Depth” blueprint is a series of nine rays, eight of which form a circle around the ninth at small angular deviations to it, each reporting a focal depth. The mean focal depth is then calculated once all depths that are statistically dissimilar to the others have been culled. This new focal depth, *Correct Depth* is the optimal depth for the user’s current focus.

With this new focal depth, the interpolation from the prior focal depth can be executed using the “Update DoF setting” blueprint shown in Figure 4.6. Firstly, the *Last Focus* variable is set to the *Focal Distance* the EGD-DoF system currently focusses at, obtained by querying the current post process settings. We then find the interpolation speed using a comparison between this *Last Focus* and *New Focus*, which is equal to the *Correct Depth* variable mentioned previously. As mentioned in Section 3.2.2, this interpolation speed is implementation dependent, and in this implementation, we deviate slightly from refocussing speeds that are physically accurate.

Depending on the ratio of *Last Focus* to *New Focus* and the absolute difference between them, three refocussing scenarios are implemented. *Interp Speed* is the variable that controls interpolation speed, and in this implementation, using d_{prev} and d_{new} to respectively refer to *Last Focus* and *New Focus*, is set to equal:

1. $\frac{d_{prev}}{d_{new}} \times 5$ when *Last Focus* \gg *New Focus*. This results in the time taken to refocus from infinite distance to a small ($\approx 1m$) distance being approximately $500ms$. This is approximate to the refocusing speed of an average adult human eye, as previously discussed.
2. $\frac{d_{prev}}{d_{new}} \times 2$ when *Last Focus* $>$ *New Focus*. In scenes dominated by smaller depth changes this provides a better visual cue for depth changes.

3. $\frac{d_{prev}}{d_{new}} \times 1$ when *Last Focus* < *New Focus*. In scenes dominated by smaller depth changes this provides a better visual cue for depth changes when focussing further away.

It should be noted that these interpolation speeds are not what the user will typically experience. As the user moves their viewport, the *Correct Depth* will change repeatedly and the interpolation values will be overwritten by each new focussing scenario. This makes estimating the actual refocussing speed very difficult and this implementation instead uses values that are considered to generate high quality output. The process by which this is done is outlined in Section 4.6.

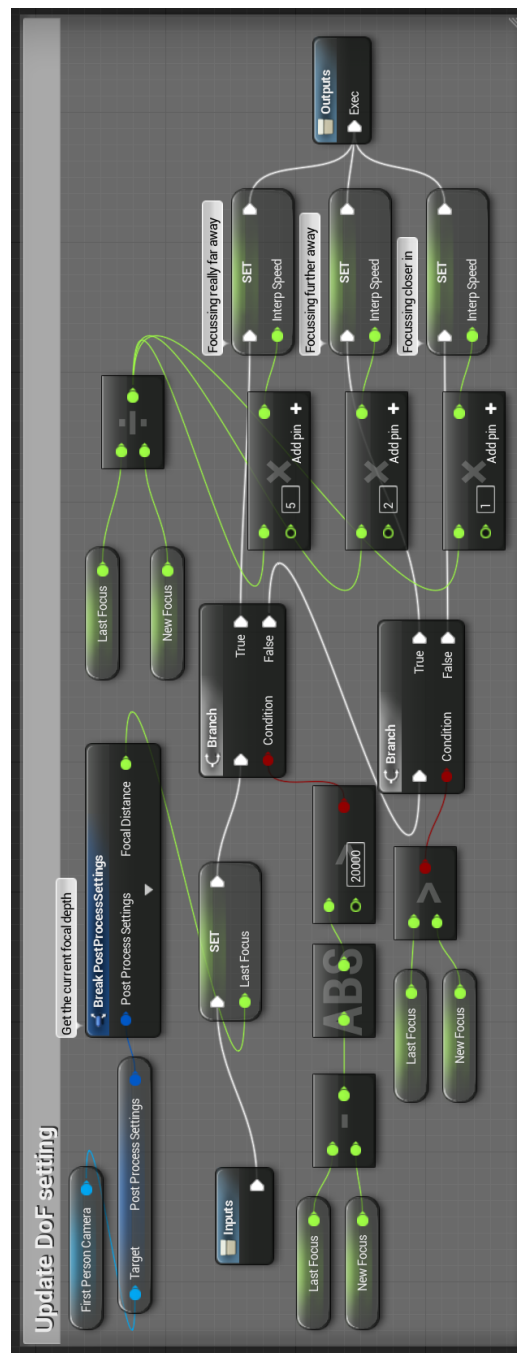


Figure 4.6: Setting the interpolation speed.

The node titled “FInterpTo” in Figure 4.7 takes *Interp Speed* as a parameter and linearly interpolates from *Last Focus* to *New Focus*, using the *Get World Delta Seconds* node to find the amount of time since the node was last triggered to work out how far the interpolation should progress. This new, interpolated depth is referred to as “Focal Distance”. Figure 4.7 is actually part of a larger blueprint titled “Update DOF Settings”, but this blueprint has been split across a multiple figures for clarity.

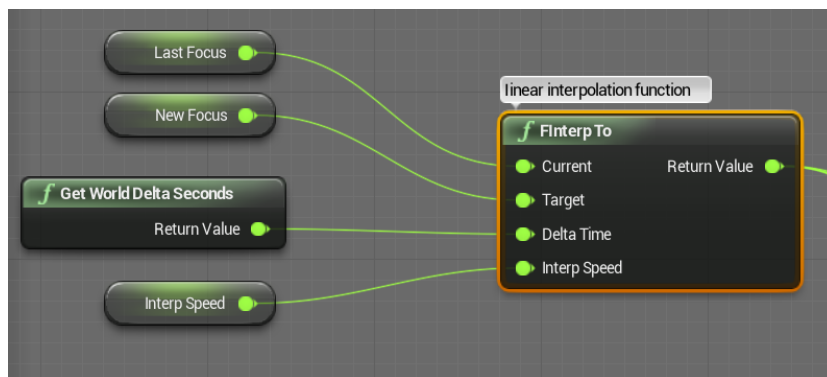


Figure 4.7: The blueprint that controls the interpolation to a new focal depth.

In implementing the EGD-DoF system in UE4, it was noticed that the blur always looked artificial when a user moved their head. If they were stationary, the blur looked good, but as soon as a movement occurred, the degree to which the scene was blurred appeared unnatural. To alleviate this, a new behaviour was introduced to the EGD-DoF system: the degree to which the scene is blurred is dependent on how long the user has focussed at a particular depth. This means that while the user is moving their head, the scene is not blurred very much in software (the Oculus is not a low persistence display, so the screen creates a blur effect when moved), but as soon as the user keeps the focus steady at a particular depth, the scene becomes progressively more blurred.

This blur amount is called the *Scale* of the blur and is controlled by the node shown in Figure 4.8. Whenever the DoF system focusses at a new depth, this node is triggered. It then repeatedly updates. Each time it updates, it samples further along the timeline shown in Figure 4.8 and returns $T0$, equal to the *Scale* of the blur. When the blur effect is turned off, this node runs in reverse to totally disable blur effects.

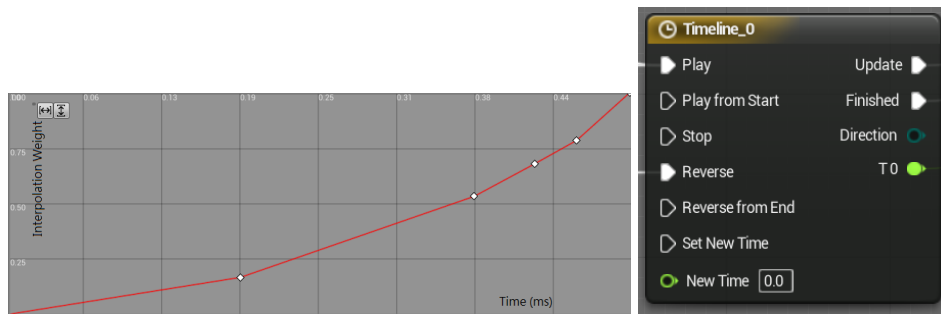


Figure 4.8: The timeline for interpolating the DoF scale factor (left) and the blueprint node that controls it (right).

The second part of the “Update DOF Settings” blueprint is shown in Figure 4.9. This is the final node in the EGD-DoF implementation as it sets up the parameters of the bokeh blur function. As inputs, this node takes *Focal Distance* and *Scale* from the blueprints and nodes discussed earlier in this section, as well as a few other parameters, discussed in Section 4.4.

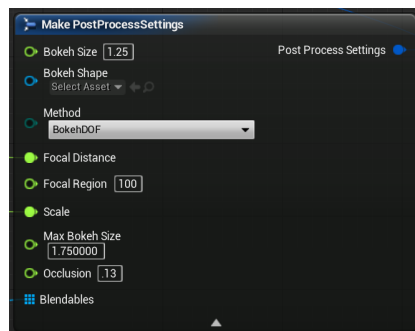


Figure 4.9: The post-process node that implements bokeh blur.

4.4 Implementing DoF blur

In UE4, implementing a post process blur is very simple. Parameters are simply passed to the node shown in Figure 4.9, which uses a reference to the *Player Camera* to apply postprocessing effects to it. A list of each of the parameters relevant to bokeh DoF that this node uses is shown in Table 4.1, alongside an explanation of each of these parameters. Note that while only bokeh DoF parameters are visible, this node takes over 200 inputs in total, including the ones that control Gaussian DoF.

Table 4.1: Parameters for the UE4 bokeh function

Parameter	Description
Bokeh Size	The typical size of the bokeh blur kernel, as a percentage of the screen width.
Bokeh Shape	The shape of the bokeh blur kernel, chosen to be circular (which is the default, explaining why it is not visible).
Focal Distance	The distance from the camera that is perfectly in focus
Focal Region	The width of the region that is in perfect focus
Scale	Scaling factor for the magnitude of the blur
Max Bokeh Size	The maximum width of the bokeh blur kernel
Occlusion	How much geometry will blur beyond its usual silhouette
Blendables	The camera the blur will be applied to the visual field of

UE4 has implementations of Bokeh and Gaussian DoF, both of which are implemented using the “Post Process Node” above and use the GPU[91].

In Section 3.4, a naive bokeh implementation was introduced. This implementation has some issues: it is slow and prone to “bleeding” (where blur from objects further away can occlude objects closer to the camera). The UE4 implementation of bokeh addresses these issues by:

1. To increase computation speed:
 - (a) The scene is sampled at quarter resolution, reducing the number of object depths that require computation by 75%
 - (b) The “focal point” is treated as a “focal region”, recognising that objects with depths close to the focal depth will have unrecognisably small CoCs that may as well not be computed.
2. To improve visual quality:
 - (a) The scene is split into three regions: near, focal and far. Respectively, the near, focal and far regions are the set of pixels with depths $< f$, $\approx f$ and $> f$. These blur regions are shown in Figure 4.10. Each of the near and far regions are blurred separately and then blended with the unblurred scene to avoid the previously mentioned bleeding issue.
 - (b) The downsampling DoF technique mentioned earlier is adaptive: downsampling will only occur in locations where it will not noticeably adversely affect the visual quality of the scene.

The result of these optimisations is a high-quality bokeh implementation that is fast: $60fps$ was achieved in our implementation on consumer hardware. As this high speed is achieved, we do not see any advantage to using Gaussian DoF over bokeh in this implementation, so the initial decision to use bokeh DoF is affirmed.

In the UE4 bokeh implementation, the radius r of the CoC for each pixel p can be approximated by the graph shown in Figure 4.10. r is defined and measured as a percentage of the screen width (further detailed

in Section 4.6), and "distance" refers to the depth disparity between the pixel in question and the current focal depth. The result of this blurring can be seen in Figure 4.11: the focal region remains sharp (in the middle of the scene) while the rest of the scene grows progressively blurred.

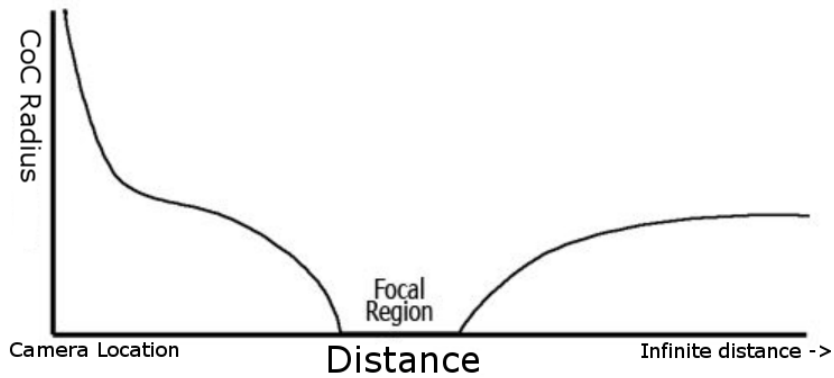


Figure 4.10: Approximation of the blur radius for different depth disparities output by the UE4 bokeh implementation.



Figure 4.11: Example of a UE4 scene with differing radii of DoF blur.

This distance based bokeh blur is the only software blur utilised in this

research: no peripheral blur was applied. This is because the optics of the Oculus achieve the same effect through what is known as variable acuity resolution (VAR): the perceived pixel density of the screen when viewed through the Oculus lenses is higher in the centre of the screen than to the sides [92]. This results in peripheral content appearing to be of a lower resolution than central content on the Oculus screen, creating an implicit blur effect.

4.5 Interacting with the scenes

In this research, an avatar with a height of $1.67m$ is used for testing the EGD-DoF implementation. This is the global gender-neutral average height of a human adult, which is therefore assumed to be the average height of a user of this system. This avatar is shown in Figure 4.12.



Figure 4.12: The player avatar, with attached camera.

It is known that higher traversal speeds in VEs increases the speed at which discomfort arises [93], hence the user's avatar is limited to a velocity of $1.4ms^{-1}$ ($5kmph$). This velocity is chosen as it is the average walking speed for a human adult. If a user's velocity has a non-zero vertical component (such as when they walk up stairs), their horizontal velocity is decreased to compensate, keeping their total velocity constant. This is done to eliminate acceleration when walking up or down stairs. If v is the avatar's velocity within the scene, v_x and v_y are the horizontal and vertical components of v respectively and $|v| = 1.4ms^{-1}$, and v_y is dependent on the change in elevation within the scene (the angle θ at which the ground is compared to the x-z plane), then v_x can be calculated using Equation 3.2:

$$v_x = \begin{cases} 1.4, & \text{if } v_y = 0 \\ 1.4\cos(\theta) & \text{otherwise} \end{cases} \quad (3.2)$$

Navigation and control is significantly more difficult in VR than in desktop viewing. As discussed in Chapter 2, unexpected motions are a significant contribution to visual discomfort, so care must be taken to ensure the player does not interact with the VE in a manner that may cause discomfort. This research had insufficient time to complete a survey of the effectiveness of different control and interaction schemes for VR, so a scheme proposed by Valve Software was employed [4]. In this implementation, the yaw of the player avatar's torso is controlled by the mouse, and the torso pitch and roll are locked. The camera roll, pitch and yaw are controlled by motions of the HMD, and movement within the scene is oriented around the torso yaw and controlled using the keyboard. This specific control scheme was selected as it was the so-called "best" scheme that Valve had proposed that was not specifically built around interactions (such as aiming) that would not be present in this system .

4.6 Selecting DoF Parameters and Constants

The maximum blur radius r_{max} and interpolation speeds mentioned previously were selected using a preliminary experiment in which a test environment was generated consisting of multiple identical objects at differing visual depths, (Fig 4.13). Through visual analysis of this scene, r_{max} was selected to be 1.75% of the screen width. DoF blur that doesn't match the expected blur of the human visual system is known to not cause visual discomfort and not decrease spatial accuracy [94], so such an estimated blur radius is not significantly disadvantageous to this system. This research had insufficient time to select these parameters using a more rigorous approach such as by modelling the human eye, due to (1) the complexity of constructing and testing such a model, and (2) the lack of explicit data on the optics used in the Oculus resulting in the requirement that the lenses parameters be investigated. A fixed inter-pupillary distance (IPD) equal to the 63.5mm (the average value for adults) is assumed. The occlusion parameter was chosen to 0.13, slightly less than the default value of 0.18, as this allowed for clearer blur effects when objects were close to the player camera. The shape of the bokeh blur kernel was chosen to be circular (the default) as this resulted in images that best corresponded to human vision.

For efficiency, we do not want rays that are cast into the scene to be cast to infinite distances: a cut-off distance is defined that creates an effective maximum ray depth. This cutoff was chosen to be 5,000,000 unreal units (equivalent to 5km) which is half of the UE4 default skybox depth. Strictly all depths above 100m require approximately the same focus in the human eye, but preliminary testing revealed that treating all rays that were longer than 100m as "infinite" did not result in a "natural-looking" result. The infinite threshold was increased until the results matched expected visual blur at 5km.

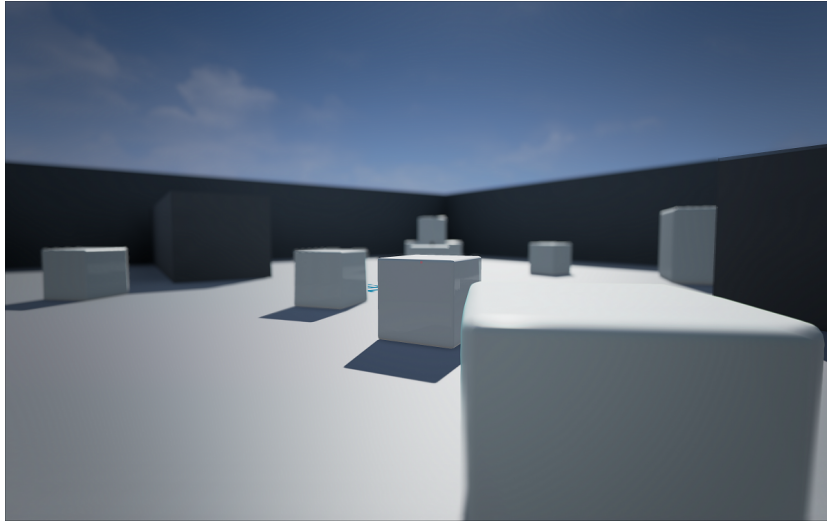


Figure 4.13: The test scene used for evaluating and selecting bokeh blur parameters.

4.7 Implementation Summary

The VE that EGD-DoF was applied to is rendered at a consistent $60fps$, and preliminary testing in the environment mentioned in the previous section shows no degradation of VE quality through the addition of DoF blur effects. The EGD-DoF system is applied to more complicated scenes and its effectiveness at reducing visual discomfort is evaluated using the processes outlined in Chapter 5.

Chapter 5

Psychophysical Evaluation

In order to evaluate the effectiveness of the EGD-DoF system in reducing visual discomfort, a psychophysical evaluation is carried out using responses to a Simulator Sickness Questionnaire (SSQ) based on [26] as a metric. This SSQ was chosen as it is accepted as the standard for subjective tests involving simulator sickness (and thus visual discomfort). In this chapter, two scenes taken from the Unreal Marketplace that were selected for testing the effectiveness of the EGD-DoF implementation are introduced. This chapter then details the process of the performed evaluation and introduces the metrics used to interpret participant results.

5.1 Setup

5.1.1 Participants

In order to find participants for this study, posters were placed around the Victoria University campus and posts on relevant social media student groups were made. An example of one of the posters used for advertising is shown in Appendix C. Potential participants were screened for the self-reported conditions listed in Table 5.1 and given information sheets (Appendix C) detailing the process of the study, following recommendations from [95]. This screening process was carried out as it is believed that effects from these conditions would have a significant impact on the results of participant sessions that would mask the impact of the EGD-DoF system under investigation. Some information was not written down

in these information sheets and was mentioned verbally as a clarification. These clarifications are shown in the second column of Table 5.1. All participants also had to complete a consent form to indicate they understood the risks and processes of the study (Appendix C).

Table 5.1: Conditions that excluded participation in the study.

Condition	Clarification
Prior experience on a HMD	Counted if prior experience exceeded 10min
Pregnancy	
Susceptibility to motion sickness	Described as "experiences motion sickness with little stimulus, i.e. when playing computer games or watching films."
Undergone recent eye surgery	Recent means within the last 12 months
Have any visual abnormalities	Excluding short/long sightedness and cosmetic changes such as heterochromia or colour blindness Examples: amblyopia, spherical and chromatic aberrations.

No compensation was offered for participation in the study, as the appeal of trying on the Oculus was considered attraction enough. These advertisements resulted in a total 21 individuals willing to participate in the study. Ethics approval was granted by the Victoria University Human Ethics Committee. Of these 21, 5 had to be excluded due to the exclusions in Table 5.1, leaving 16 participants, with ages ranging from 18 to 25 years old. Three participants were female, and all participants were current or prior students of Victoria University of Wellington. This does introduce a limitation to the results of the study: visual performance is known to change with age, so our findings have primary relevance to

young adults only and care should be taken when extrapolating them to other ages groups.

5.1.2 Testing hardware

Participants completed the study on a machine running Windows 7 with 8GB RAM, a 3.6GHz Intel quadcore CPU and a NVIDIA GTX 770 GPU with an attached Oculus Rift DK1 HMD. An example picture of the experimental set-up is shown in Figure 5.1. This machine was chosen as it is powerful enough to reliably achieve 60fps on the scenes rendered during the evaluation, without being unacceptably expensive (thus being a reasonable approximation of consumer hardware). In accordance with the objectives of this research, no external testing hardware was utilised in this evaluation.

5.1.3 Testing scenes

One of the advantages of implementing the EGD-DoF system in the Unreal Engine is that the Unreal marketplace offers a large variety of pre-built scenes, some of which can be freely accessed and modified. Two such scenes from the marketplace were adapted for use in this research: “Sun Temple” and “Landscape Mountains”, which are shown in Figures 5.2 and 5.3 respectively. Larger screenshots of each scene are shown in Appendix B.

These immersive 3D scenes were selected to broadly cover features and environmental factors common to game and film applications, as summarised in Table 5.2.



Figure 5.1: A participant during the user study.

Table 5.2: Comparison of factors in each scene

Sun Temple	Landscape Mountains
Even walking surfaces with stairs	Uneven walking surfaces with inclines
Rooms at a variety of light levels	Consistent lighting
Mostly objects at small focal depths	Mostly objects at long focal depths
Windows (to look out from)	Bridges (to look down from)
Only enclosed areas traversable	Only open areas traversable

5.2 Stimuli

There were four possible broad types of stimulus for this evaluation. Participants could experience the selected scenes:



Figure 5.2: Screenshots of the Sun Temple scene with DoF enabled (top) and disabled (bottom).

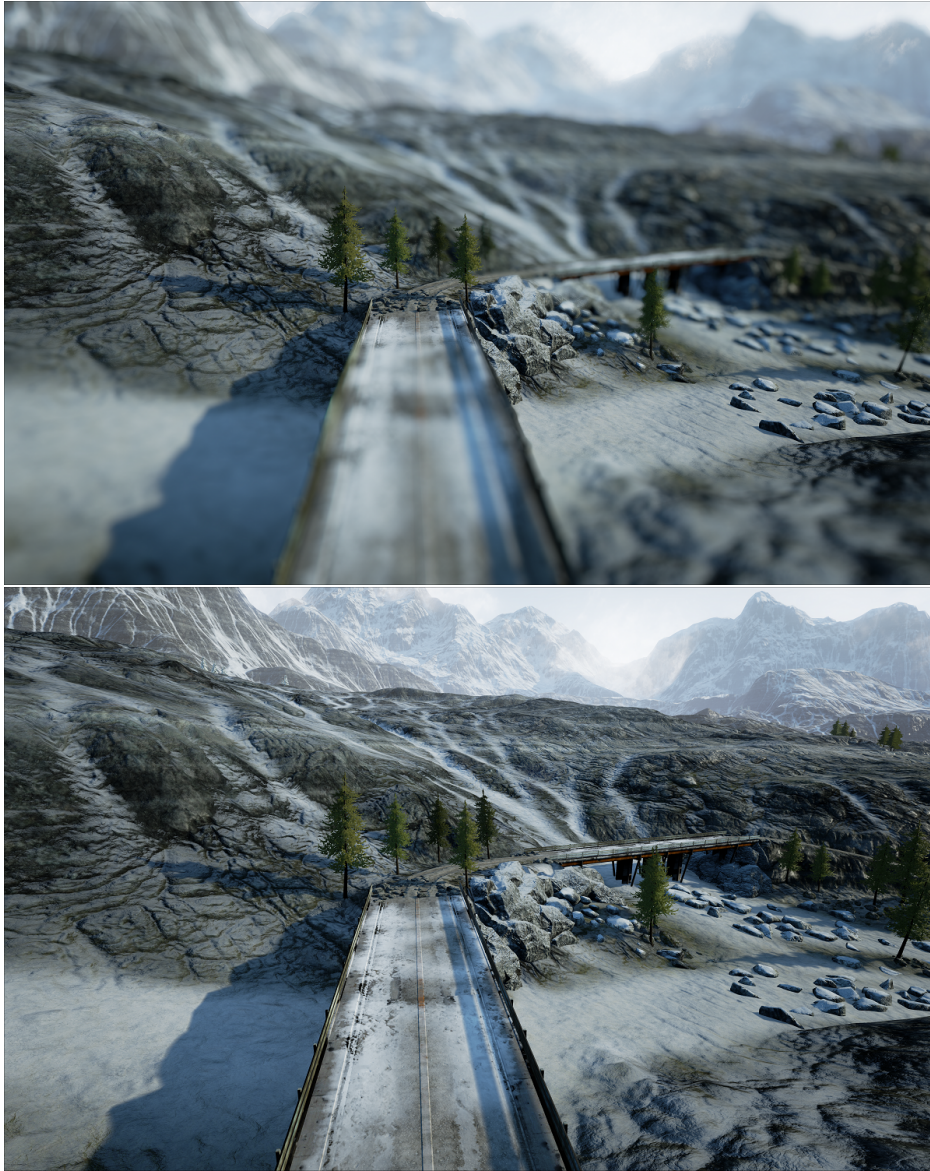


Figure 5.3: Screenshots of the Landscape Mountains scene with DoF enabled (top) and disabled (bottom).

1. From a static position, either with a fixed or movable camera orientation.
2. “On the rails”, with player camera moving along a predefined path, similar to a rollercoaster.
3. With complete control of their position and orientation, but with guidance or a set of tasks to complete.
4. Freely, without any input or restriction on their movements.

Method (3) was selected as it is known that taking control away from someone in a VE is a cause of discomfort, and giving users total freedom could result in too large a spread of actions: different users could explore the scenes in significantly different ways, introducing a difficult to quantify variable to the evaluation. In each scene the users were thus given a small task they could complete: In the Sun Temple, they were instructed to “go have a look around, check out all the rooms”, and in the Landscape Mountains scene they were instructed to “go find the crashed UFO at the top of a mountain”.

5.3 Procedure

Each of the Sun Temple and Landscape mountains scenes were shown to participants.

The study involved two sessions per participant, each on sequential working days. In the first session, the participant was shown the two scenes in a random order, with DoF blur applied to one randomly chosen scene. In the second session, the DoF blur was applied to the other scene, and the ordering of the scenes was reversed. The number of participants whose first session had DoF blur enabled is equal to the number of participants whose first session had DoF blur disabled. At the beginning of each session, participants were informed that they were allowed to withdraw

from the experiment at any time if they felt too sick. No specific definition was given for how sick “too sick” refers to: it was a subjective individual decision. Participants with glasses were given the option to wear them inside the Oculus if they could fit, otherwise the appropriate lenses were put into the Oculus to compensate.

A SSQ consisting of the 23 questions in Table 5.3 was used in the study. For each question, the participant is verbally asked their current experience of a symptom using a 5 point Likert scale (ranging from “None”, indicating no presence of that particular symptom, to “Severe”, indicating severe or traumatic presence). The SSQ responses resulted in an 18 part measurement of subjective discomfort across specific symptoms.

Table 5.3: SSQ symptoms and the corresponding discomfort categories, from [26].

Symptom	Categories
General discomfort	Nausea, Oculomotor
Fatigue	Oculomotor
Boredom	None
Drowsiness	None
Headache	Oculomotor
Eyestrain	Oculomotor
Difficulty Focusing	Oculomotor, Disorientation
Blurred Vision	Oculomotor
Salivation increase	Nausea
Dry Mouth	None
Sweating	Nausea
Nausea	Nausea, Disorientation
Difficulty Concentrating	Nausea, Oculomotor
Fullness of head	Disorientation
Anxiety	None
Dizziness	Disorientation
Vertigo	Disorientation
Visual Flashbacks	None
Awareness of Breathing	None
Stomach Awareness	Nausea
Loss of appetite	None
Desire to move bowels	None
Confusion	None

In total, during each session, participants were asked to complete 4 SSQs. The exact structure of each session and timings of each SSQ are as follows:

1. SSQ #1.
2. Participant puts on the Oculus, and adjusts its physical settings to be comfortable.
3. The nature of the control scheme is explained.

4. Participant gets a minute to familiarise themselves with the control scheme.
5. Participant explores scene 1 for 15min.
6. SSQ #2.
7. Participant is instructed to close their eyes while scene 2 loads (to avoid the illusion of motion).
8. Participant explores scene 2 for 15min.
9. SSQ #3.
10. Participant takes the Oculus off and relaxes for 15min.
11. SSQ #4.

5.4 The Metric for Determining Visual Discomfort

Not all of the symptoms that were investigated during each session are used for quantifying participant sickness. The *Boredom*, *Drowsiness*, *Anxiety* and *Confusion* symptoms were used only to give an estimation of a user's mental state during a session. The relationship between these responses and participant discomfort is explored in Chapter 6. *Loss of appetite* and *Desire to move bowels* were not considered a factor of participant discomfort because it was not possible to control external factors such as time since last meal, which would significantly affect these symptoms. *Dry Mouth* was only asked as a control question: a participant who indicated they had both increased salivation and a dry mouth may not be giving accurate or truthful answers. *Awareness of Breathing* was intended as a serious question (unusual breathing implying discomfort) but was consistently considered a joke by participants and thus useful information was

not obtainable.

Each SSQ completed thus resulted in 14 numbers (each in the range [1, 5], as mentioned in the previous section) that correspond to a participant's subjective experience of discomfort symptoms. When the SSQ was first introduced, it was recommended that only post-exposure data should be used for comparing participant discomfort [26] (SSQ #3 in this experiment). However, the original SSQ was tested on a standardised population in terms of participant fitness and health (military personnel in flight simulators), allowing a common initial state to be defined. In contrast, this research involves participants who are student volunteers with varied health and fitness levels. As a result, it is proposed that comparing only post-exposure SSQ data is insufficient for analysis, as no common initial state can be established in this more varied population. Other studies follow this comparative approach, such as [58].

To establish a metric for quantifying the discomfort caused by exposure to the VE, we define the following terms:

- $N_{1..6}$, $O_{1..7}$ and $D_{1..5}$: the raw results from a SSQ, sorted into each discomfort category (nausea, oculomotor and disorientation respectively). Since four symptoms count towards more than one category, a total of 18 unique values result.
- N_C , O_C and D_C , the total discomfort that a participant is experiencing at the time of a given SSQ for a particular discomfort category.
- S_C , the “current discomfort measure”: the total discomfort that a participant is experiencing at the time of a given SSQ.
- S_T , the change in total discomfort that a participant experiences after 30 minute of HMD exposure (i.e., between SSQ#1 and SSQ#3).

Using the numerical “scaling factors” from [26], each of N_C , O_C and D_C can be defined using Equation 5.1. While this research is using a comparative sickness measure in contrast to the absolute measure in [26], we consider their scaling factor to still be appropriate for use here.

$$N_C = \Sigma N \times 0.954, \quad (5.1a)$$

$$O_C = \Sigma O \times 0.758, \quad (5.1b)$$

$$D_C = \Sigma D \times 1.392 \quad (5.1c)$$

A participant’s S_C score can be calculated using the sum of the discomfort totals for each category (Equation 4.2):

$$S_C = N_C + O_C + D_C \quad (4.2)$$

The superscript x is used (S_C^x or N_C^x) to represent an individual category or total discomfort score calculated for a given SSQ. For example, S_C^1 is the total discomfort for a participant’s first SSQ, asked prior to exposure to the HMD.

S_T is calculated using Equation 4.2:

$$S_T = S_C^3 - S_C^1 = (N_C^3 - N_C^1) + (O_C^3 - O_C^1) + (D_C^3 - D_C^1) \quad (4.2)$$

This results in a value that expresses the increase in severity of each symptom that each participant experienced after 30min of HMD exposure. As an example, Table 5.4 outlines the results calculations of S_T for a fictitious participant, with their SSQ results split into each of the three discomfort categories.

Table 5.4: A fictitious participant's SSQ results and S_T score.

Symptom	SSQ#1	SSQ #3	SSQ#3 - #1
Nausea			
General discomfort	1	3	2
Salivation increase	1	2	1
Sweating	1	1	0
Nausea	1	2	1
Difficulty Concentrating	2	2	0
Stomach Awareness	1	1	0
$N_C^1 = 7 \times 0.954 = 6.678$ $N_C^3 = 11 \times 0.954 = 10.494$			
Oculomotor			
General discomfort	1	3	2
Fatigue	2	2	0
Headache	3	3	0
Eyestrain	2	2	0
Difficulty Focusing	2	3	1
Blurred Vision	1	1	0
Difficulty Concentrating	2	2	0
$O_C^1 = 11 \times 0.758 = 8.338$ $O_C^3 = 16 \times 0.758 = 12.128$			
Disorientation			
Difficulty Focusing	2	3	1
Nausea	1	2	1
Fullness of head	1	1	0
Dizziness	1	1	0
Vertigo	1	1	0
$D_C^1 = 6 \times 1.392 = 11.136$ $D_C^3 = 8 \times 1.392 = 13.92$			
Total			
$S_C^1 = 26.152$ $S_C^3 = 36.542$ $S_T = S_C^3 - S_C^1 = 10.39$			

S_T is used as the primary measure for determining visual discomfort

arising on a HMD in this research, with greater S_T scores indicating a greater increase in discomfort. Chapter 6 evaluates the results drawn from this data.

Chapter 6

Results and Analysis

In the previous chapter a psychophysical study for evaluating the effectiveness of the EGD-DoF system was introduced, and a total sickness measure, S_T was defined as the primary metric for determining visual discomfort. In this chapter, the results of the psychophysical evaluation are introduced and analysed. Some independent variables and important results for individual symptoms (including those not used for calculating S_T) are presented and discussed. The changes in each of the three discomfort categories when the EGD-DoF system is enabled are analysed and finally the overall conclusions are presented regarding the effectiveness of the EGD-DoF system at reducing discomfort.

6.1 Withdrawals

Participants were allowed to withdraw from each session at any time if they felt overwhelmingly sick. Of the sixteen participants who took part in the study, five chose to withdraw during one of their sessions. All five of these participants were willing to answer a SSQ after their withdrawal. Two participant's results were discarded due to an error in recording their results.

By calculating S_T for each participant, the frequency distribution shown in Figure 6.1 is generated. Results in blue indicate when the EGD-DoF system was enabled, orange represents when it was disabled, and stars indicates sessions where a participant withdrew.

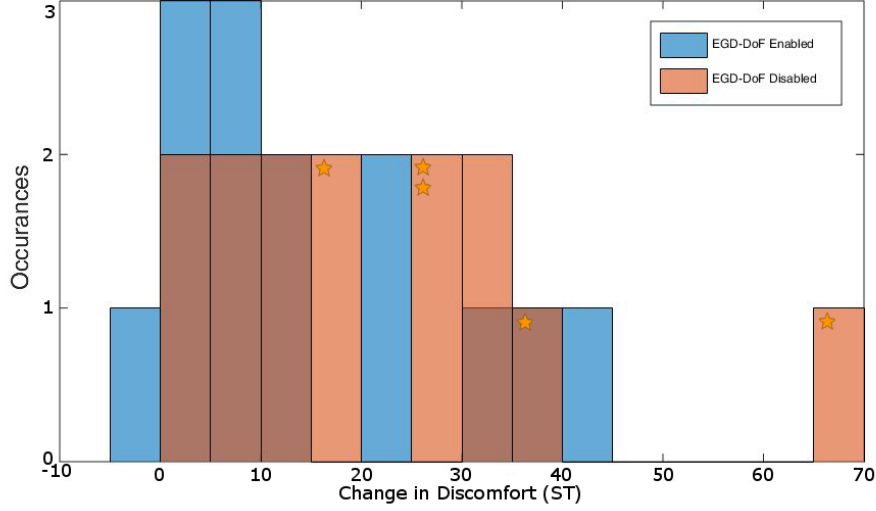


Figure 6.1: Frequency distribution of S_T scores. Stars indicate sessions where a participant withdrew. Note that this diagram shows two overlapping distributions, so brown results correspond to two results in the same ‘bin’: one from each of the EGD-DoF enabled and disabled distributions.

Figure 6.1 illustrates that all of the withdrawals occurred in sessions where EGD-DoF was not enabled, and the S_T scores for these sessions are notably high. There are however, sessions where participants recorded S_T scores higher than these withdrawal sessions.

By calculating S_C for each participant (the discomfort measure post-HMD exposure) we obtain the frequency distribution shown in Figure 6.2. Investigating S_C (the absolute value of sickness after 30 minutes) allows for a validation of the selection of S_T (the change in sickness after 30 minutes) as the primary measure for calculating total sickness: if the S_C scores show all withdrawal sessions are separable from non-withdrawal sessions, or new trends are revealed, then the S_T measure can be concluded to be masking useful information and thus ill-suited to being the primary measure of discomfort.

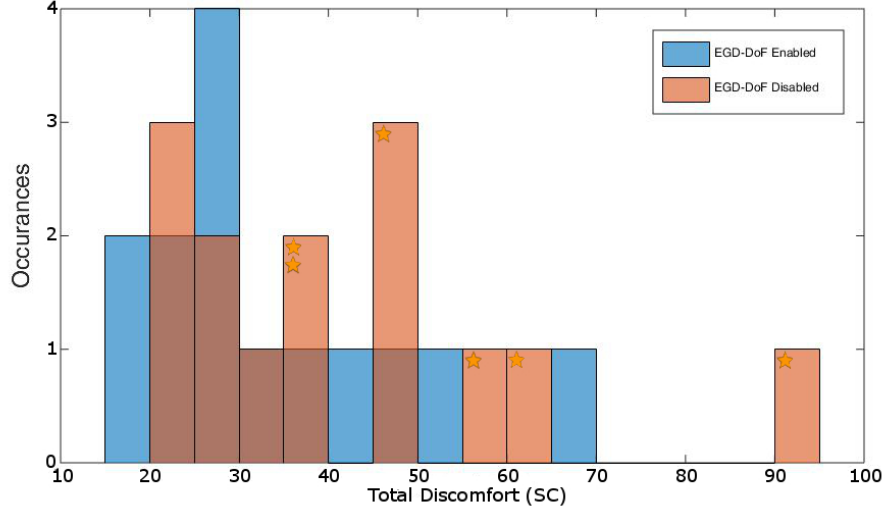


Figure 6.2: Frequency distribution of S_C scores. Stars indicate results where a participant withdrew.

Again, we see results where participants who did not withdraw (non-shaded results) during their session had higher sickness (in this case, S_C) scores than participants who did withdraw (shaded results). Evidently, a participant's change in discomfort and total discomfort scores alone cannot be used to determine if the participant will withdraw or not: individual factors such as discomfort tolerance allow some participants to withstand both a larger increase in, and higher total discomfort than others. This means that the S_T measure does not mask a trend visible in S_C , validating its selection as the primary sickness measure.

In Figure 6.3 the S_T results for each session are shown, in the order in which the sessions were completed. Two results are present for each participant (which usually are not co-located, as multiple participants completed the study each day). Of particular interest are the sessions indicated in red, which are the sessions in which a participant withdrew: there is an

unusual cluster of withdrawals on one particular day. These sessions occurred on an unusually warm day. Given that visual discomfort severity is known to increase as the ambient temperature increases, this could account for these withdrawals [96]. As further discussed in Section 6.3, this highlights the requirement that these external factors are controlled during psychophysical evaluations.

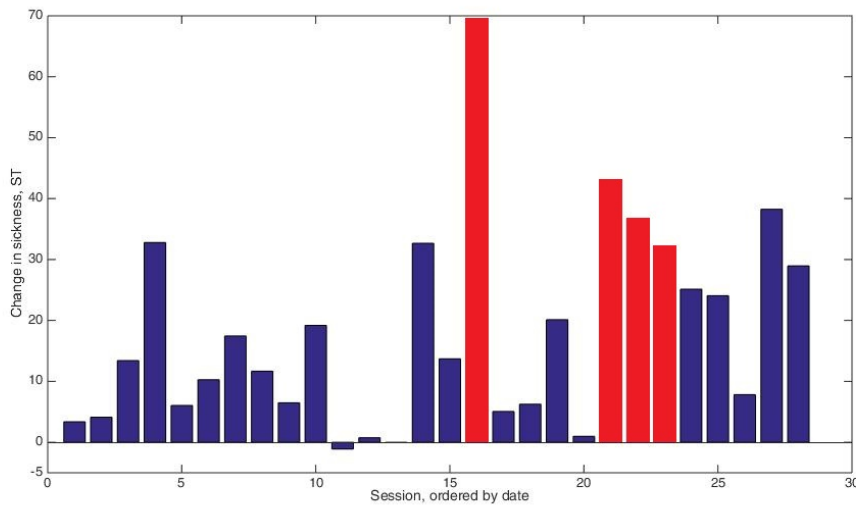


Figure 6.3: S_T results for each session, ordered by the date the session was completed.

6.2 Individual Symptom Results and Analysis

As mentioned in Section 5.4, not all of the symptoms in our modified SSQ are used to directly evaluate discomfort, and some are otherwise not used due to independent variables such as time since last meal. This resulted in a total of 14 unique symptoms that were used to calculate total sickness. Before summing these individual symptoms to calculate S_T , it is interesting to directly compare the average change in the severity of each

symptom between the two sessions for each participant. Figure 6.4 shows the average increase in severity for each of these 14 symptoms for each session: yellow for when the EGD-DoF system was not enabled and blue for when the system was enabled.

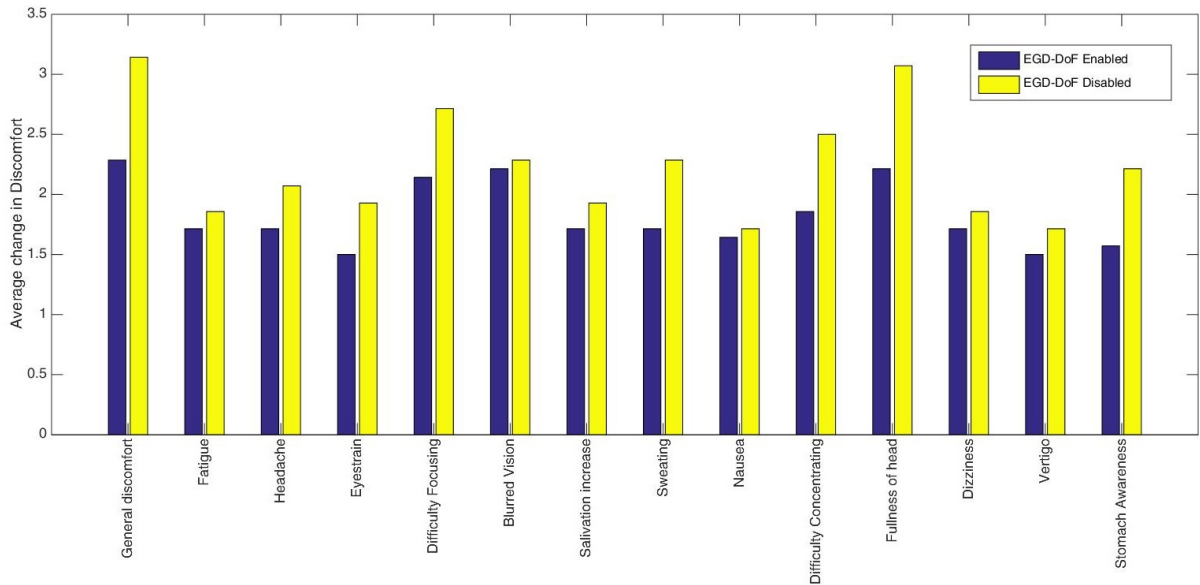


Figure 6.4: A comparison between averages of participant symptom severity with DoF blur enabled and disabled, for symptoms used in S_T calculation.

It is apparent upon visual analysis of Figure 6.4 that on average, participants experienced smaller increases in symptom severity when the EGD-DoF system was enabled. However, when EGD-DoF was enabled, all symptoms still show a 1.5 to 2 point increase in severity indicating that while this system reduces the severity of visual discomfort symptoms, it does not eliminate them.

When the symptoms that were not used for calculating S_T are investigated (Figure 6.5) it is noticeable that every one of these “mental” symp-

toms, as they were introduced in Section 5.4 has a greater (or approximately equal) increase in severity when the EGD-DoF system is not enabled. Participants on average show no increase in anxiety when EGD-DoF is enabled, and an average ≈ 0.4 point increase when it is not. This illustrates the importance of perceptually optimised rendering in VR: an increase in anxiety is not a desirable response for simple exercises such as walking in a supposedly stress-free environment. No significant change in participant boredom is noticed, but an increase in both confusion and drowsiness is. The relative increase in confusion is likely correlated with the increase in disorientation discomfort discussed in Section 6.4. The increase in drowsiness is proposed to be a side-effect of decreased immersion in the scenes: as previously discussed, unoptimised rendering results in decreased presence and immersion in VEs. It follows that in a scene with lesser immersion, a participant would feel less engagement with the VE and become drowsy.

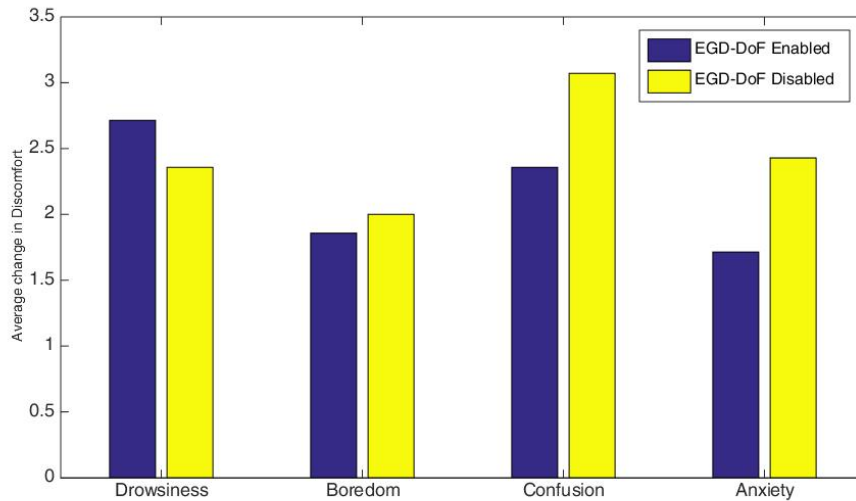


Figure 6.5: A comparison between average of participant symptom severity with DoF blur enabled and disabled, for symptoms not used in S_T calculation.

6.3 Independent Variables

Since it is known that visual discomfort decreases with repeated exposure for most people, it would be understandable if S_T scores for participant's second sessions were lower than each of their first sessions. In order to attempt to mitigate any effect this acclimatisation would have on results, we decided to have exactly half of the participants start with EGD-DoF enabled, and have half start without. From this, the paired graphs shown in Figures 6.7 and 6.6 are generated. In Figure 6.6, the blue line represents the EGD-DoF system being enabled, and the orange shows results when it was disabled. In Figure 6.7 the same participant order applies, and blue shows a result from a participant's first session, and orange shows their second sessions.

If the VR acclimatisation that occurs within 30 minutes is significant, then it is expected that a trend would arise in Figure 6.7 that is not present in Figure 6.6: as it is, 6 of the 14 results show a lower S_T score in the first session compared to the second, and 11 of the 14 results show lower S_T scores in sessions when EGD-DoF was enabled. It is thus concluded that the acclimatisation effect from 30 minutes of exposure is not sufficient to mask the effect of the EGD-DoF system.

In Section 6.1 the correlation between an unusually warm day and withdrawals from the study was introduced. This is an example of an independent environmental factor that should be controlled during user studies. Although we were assured that the chosen location for this research would be environment controlled, the A/C did not appear to be working fully on this day.

Another factor that was not possible to control was the spread of information about the study, and the increase in knowledge among participants

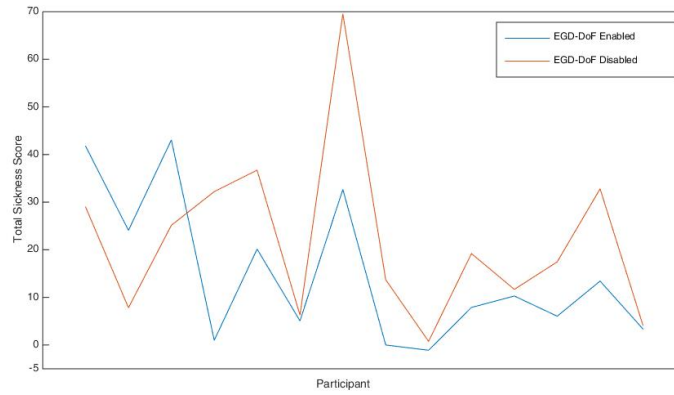


Figure 6.6: S_T scores, comparing EGD-DoF enabled/disabled.

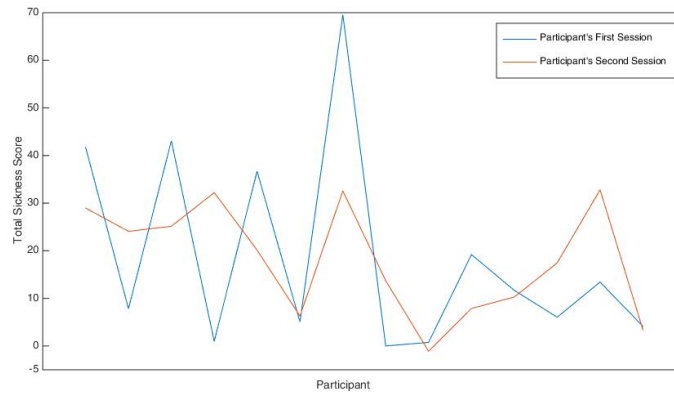


Figure 6.7: S_T scores, comparing participant's first and second sessions.

that exposure to PI-VR HMDs tends to result in visual discomfort. This was exemplified by a participant who mentioned “[my friend] undertook your study and thought it was super cool, so he mentioned I should try it as well and see how sick I got in comparison”. This sort of prior knowledge can drastically alter a participant’s perception of the experiment and their results, but this research did not have an effective way to discourage or account for such biases.

6.4 Discomfort Categories

Using the formula introduced in Equation 5.1, the total discomfort for each of the discomfort categories can be calculated for each SSQ a participant completed. Taking the average of all participants, and finding the difference between the SSQ #1 and SSQ #3 scores gives ΔN_C , ΔO_C and ΔD_C : the increase in symptom severity for each of the three discomfort categories, tabulated in Table 6.1.

Table 6.1: Comparison of mean discomfort across each of the discomfort categories.

Category	EGD-DoF			
	Mean		Std Dev	
	Disabled	Enabled	Disabled	Enabled
Nausea	8.43	5.34	6.52	6.11
Oculomotor	5.79	4.17	5.24	4.62
Disorientation	7.65	4.97	6.86	4.71

It apparent that the EGD-DoF system decreased the average mean per-category discomfort for participants in this study. Considering the distribution of results in Figure 6.1, it is apparent that there was a significant variation in the magnitude of participant discomfort, which is reflected in the large standard deviations in Table 6.1. This indicates that the magnitude of the discomfort each participant experienced is strongly influenced by personal factors: some participants experience very little discomfort where others experience significantly more.

Figure 6.8 shows a more thorough overview of each discomfort category, showing the spread of participant discomfort scores across each of

the three categories. By visual inspection, both the oculomotor and disorientation discomfort categories show small movements of the median compared to the nausea category. Using the analysis method employed in [48] a Wilcoxon signed-rank test is performed on participant discomfort on each of the three categories in order to establish whether the decrease in mean discomfort is statistically significant. No statistically significant difference was found for oculomotor discomfort ($Z = -1.0133, p = 0.15625$), but statistically significant results were found for both disorientation ($Z = -1.8043, p = 0.03593$) and nausea ($Z = -1.6474, p = 0.04947$) discomfort. It is thus concluded that the presence of the EGD-DoF system is effective in reducing both disorientation and nausea contributions to visual discomfort.

6.5 Overall Visual Discomfort

Summing the ΔN_C , ΔO_C and ΔD_C values used in the previous section gives S_T , the change in total sickness measure. The average S_T score across participants when the EGD-DoF system is enabled is ($\mu = 14.48$, $\sigma = 14.78$), compared to ($\mu = 21.88$, $\sigma = 17.93$) without. The distribution of S_T scores is shown in Figure 6.9.

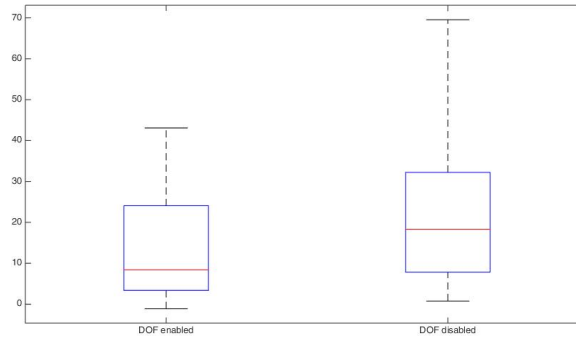


Figure 6.9: Total sickness score for participants with (left) and without (right) DoF enabled.

A Wilcoxon signed-rank test is performed on S_T in order to establish whether the decrease in S_T is statistically significant. The results do represent a statistically significant reduction in mean discomfort ($Z = -1.7264$, $p = 0.04182$). We thus conclude that the EGD-DoF system is effective in reducing visual discomfort for general exploration activities in a VE on a PI-VR HMD. This conclusion is applicable to young adults primarily: it has not been investigated whether this result equally applies to young children or older adults.

Participants in the study also expressed a preference for the sessions where the EGD-DoF system was enabled. Verbal comments made after the study include: “I found the [session with EGD-DoF enabled] much

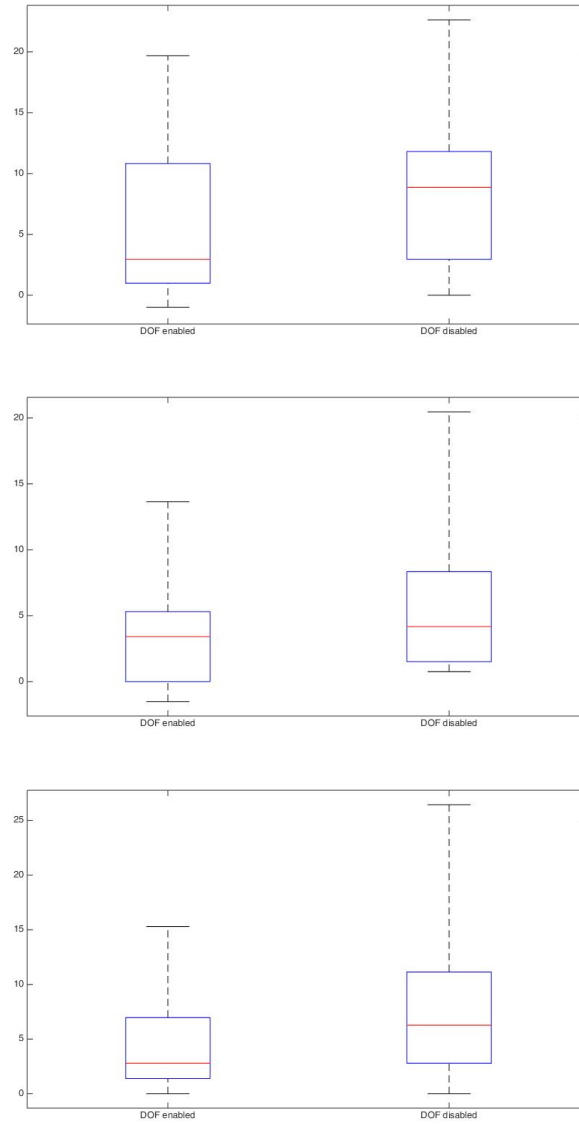


Figure 6.8: Change of sickness measure for each of the three discomfort categories: Nausea (top), Oculomotor (centre), Disorientation (bottom).

better than the [session without]: in the [session without] I felt very sick and felt a lot of eyestrain” and “The [session with EGD-DoF] was much more comfortable”.

6.6 Summary of Results

Overall participant discomfort (S_T) decreased from $\mu = 21.88$ to $\mu = 14.48$, a reduction of $\approx 34\%$. This reduction in discomfort is statistically significant, with ($Z = -1.7264, p = 0.04182$). A decrease in mean discomfort was observed in each of the three discomfort categories: nausea discomfort was reduced by $\approx 36\%$, oculomotor discomfort by $\approx 28\%$ and disorientation by $\approx 35\%$. Of these results, only the nausea and disorientation decreases were statistically significant: ($Z = -1.6474, p = 0.04947$) and ($Z = -1.8043, p = 0.03593$) respectively.

Chapter 7

Discussion and Conclusion

In this research, EGD-DoF was proposed to be an effective software solution for reducing the severity of visual discomfort on users of HMDs. In this chapter, a review of an EGD-DoF implementation in UE4 is presented alongside a summary of a user test performed to evaluate the effectiveness of said system at reducing visual discomfort. Limitations of the EGD-DoF implementation are summarised and possibilities for future work that either addresses these limitations or expands on the results of this research are explored.

7.1 Summary

In Chapter 2 a series of minimum thresholds for achieving comfortable, immersive viewing on PI-VR HMDs were introduced. As consumer hardware is currently unable to meet these thresholds, the majority of users of HMDs experience visual discomfort during usage. Existing software methods for reducing visual discomfort do not sufficiently compensate for these hardware limitations, so we concluded that a novel software solution was required.

A DoF system that estimates the direction of a user's gaze when using a HMD (EGD-DoF) and uses this estimate to dynamically blur a VE was proposed. Such a system was theorised to reduce the severity of visual discomfort caused by HMDs by affecting a user's visual behaviours when on a HMD.

In typical viewing conditions in reality, objects outside the focal plane are blurred. This contrasts with content typically rendered in VEs, where the majority of a scene is sharp even when a user's focus is at a depth such that blurring effects would be expected. In Section 2.3, an overview of prior research concluded that "expectation mismatch" is a cause of visual discomfort in VEs. By introducing out of focus blurring effects in a VE, the EGD-DoF system allows the VE to approximate the blur behaviours of reality. We expect this to reduce the contribution of "expectation mismatch" to visual discomfort.

Prior research has shown that DoF has a strong influence on the "allocation of user attention" on a screen: people typically focus less on defocussed regions. In a stereoscopic scene we expect that such blur effects would thus drive user attention away from content outside the focal plane. As a result, EGD-DoF limits the range of depths that a user focusses on at any given time, reducing the strain on the visual system caused by constantly changing vergence demands. This blur would also reduce the region of the HMD screen that a user's eyes would focus on, reducing the burden on the visual system in a HMD.

As DoF blur masks high frequency spatial data, the EGD-DoF system is also theorised to decrease the contribution of the accommodation-vergence conflict towards visual discomfort on PI-VR HMDs.

EGD-DoF is implemented in UE4 in order to produce a game environment for evaluating its effectiveness in reducing visual discomfort that users are familiar with. The system was designed to keep the centre of the virtual viewport of the Oculus in focus at all times as the user's gaze was assumed to fixate at the centre. A psychophysical evaluation was undertaken to evaluate the effectiveness of EGD-DoF at reducing discomfort. EGD-DoF was found to be effective in reducing the increase in discom-

fort that participants experienced after 30 minutes of exposure, with a 34% decrease in our sickness measure S_T observed. Using a Wilcoxon signed-rank test, it was found that this reduction in visual discomfort is statistically significant at $p < 0.05$.

7.2 Limitations and Future Work

As discussed in Section 5.1, there was a lack of variety in the participants in the study performed to gauge the effectiveness of the EGD-DoF system. Specifically, the participants were all of similar ages, and were overwhelmingly male. A simple extension to this research would be to perform the same study with a wider range of participant ages.

The parameters used in this implementation of EGD-DoF were primarily selected using a visual analysis of a sample scene according to the qualitative metric of “how natural content appeared”. Further research could be done into the visual behaviours of the human eye to quantitatively evaluate the correct blur parameters for the human visual system, given the optical configuration of the Oculus.

Viewing parameters such as the IPD, walking speed and height of the user were assumed to be equal to the human average values. This is a potential issue for users who differ greatly from the average values: for example, a very tall person may be significantly perturbed by a camera that makes them feel shorter than formal. Further studies could include participant customisation to allow the VE to be tailored to each individual, as recommended by the developers of the Oculus [46]. Further experiments could also allow for differing control schemes, as the mouse and keyboard system used is not ubiquitous, and no research has been published to show its advantages over other control systems.

Further research could also attempt to refine such a system by investigating different methods of gaze estimation when eyetracking is not available. Possibilities include using optical flow and intelligent compensation for eye movements during head motions.

The quantitative effect of DoF blurring on user focus could also be explored, in order to verify the theory behind the implementation of the EGD-DoF system.

7.3 Conclusion

A DoF system that estimates a user's gaze direction and applies a blur accordingly (EGD-DoF) was proposed as a solution for minimising visual discomfort on HMDS. By discouraging potentially discomforting focusing, this system alleviates the severity of the accommodation-vergence conflict in the visual system, reducing visual discomfort. An implementation of such a system was built in a commercial game engine.

Psychophysical experiments to evaluate the effectiveness of the EGD-DoF blur on visual discomfort on immersive HMDs such as the Oculus found that visual discomfort reduces significantly when blurring is enabled. A statistically significant reduction in both disorientation and nausea discomfort was observed, leading to a reduction in total sickness (S_T) of approximately 30% when EGD-DoF is enabled. Future applications of this result include more comfortable and realistic viewing of 3D stereoscopic content such as film and games on PI-VR HMDs.

Bibliography

- [1] J. P. Rolland and O. Cakmakci, "The past, present, and future of head-mounted display designs," 2005.
- [2] J. P. Rolland and H. Hua, "Head-mounted display systems," pp. 1–13, 2005.
- [3] Kickstarter Inc, "Oculus Rift: Step into the Game." <https://www.kickstarter.com/projects/1523379957/oculus-rift-step-into-the-game>. Accessed: 2015-01-27.
- [4] J. Ludwig, "Lessons learned porting Team Fortress 2 to Virtual Reality," 2013.
- [5] M. Lombard and T. Ditton, "At the heart of it all: The concept of presence," *Journal of Computer-Mediated Communication*, vol. 3, no. 2, pp. 0–0, 1997.
- [6] I. P. Howard and B. J. Rogers, *Seeing in Depth, vol. 2 Depth Perception*. Thornhill, Ontario, Canada: Oxford University Press, 2002.
- [7] M. McCauley, *Research issues in simulator sickness: Proceedings of a workshop*. Washinton, D.C: National Academy Press, 1984.
- [8] K. M. Stanney, K. S. Hale, N. I., and R. S. Kennedy, "What to expect from immersive virtual environment exposure: Influences of gender, body mass index, and past experience," *Human Factors*, vol. 45, pp. 504–520, Fall 2003.
- [9] R. S. Kennedy, K. M. Stanney, and W. P. Dunlap, "Duration and exposure to virtual environments: Sickness curves during and across sessions," *Presence: Teleoperators and Virtual Environments*, vol. 9, no. 5, pp. 463–472, 2000.

- [10] R. Welch, "Adapting to virtual environments," *Handbook of virtual environments*. Lawrence Erlbaum Associates, Mahwah, NJ, pp. 619–636, 2002.
- [11] T. Forsyth, "Developing VR experiences with the oculus rift." GDC 2014, 2014.
- [12] H. L., "Stereoscopic-television apparatus for individual use," Oct. 4 1960. US Patent 2,955,156.
- [13] K. Kiyokawa, "An introduction to head mounted displays for augmented reality," *Emerging Technologies of Augmented Reality* (Ed. Haller, Thomas and Billinghamurst), 2008.
- [14] R. Kalawsky, *The science of virtual reality and virtual environments*. Addison-Wesley Longman Publishing Co., Inc., 1993.
- [15] I. E. Sutherland, "A head-mounted three dimensional display," in *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, pp. 757–764, ACM, 1968.
- [16] Oculus VR LLC, "Oculus Rift - Virtual Reality Headset for Immersive 3D Gaming." <https://www.oculus.com/rift/>. Accessed: 2014-12-28.
- [17] Oculus VR LLC, "The All New Oculus Rift Development Kit 2 Virtual Reality Headset." <https://www.oculus.com/dk2/>. Accessed: 2015-02-10.
- [18] L. Jacobson, "Welcome to the virtual world," *On the cutting edge of technology*, pp. 69–79, 1993.
- [19] L. Brill, "Metaphors for the traveling cybernaut (virtual reality)," *Virtual Reality World*, vol. 1, no. 1, 1993.

- [20] P. J. Costello, *Health and safety issues associated with virtual reality: a review of current literature*. Citeseer, 1997.
- [21] M. Slater, "Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 364, no. 1535, pp. 3549–3557, 2009.
- [22] R. COULTER, L. SALAND, T. P. CAUDELL, T. E. GOLDSMITH, and D. C. ALVERSON, "The effect of degree of immersion upon learning performance in virtual reality simulations for medical education," *Medicine Meets Virtual Reality 15: In Vivo, in Vitro, in Silico: Designing the Next in Medicine*, vol. 125, p. 155, 2007.
- [23] M. Schuemie, M. Bruynzeel, L. Drost, M. Brinckman, G. De Haan, P. Emmelkamp, and C. Van der Mast, "Treatment of acrophobia in virtual reality: A pilot study," in *Conference Proceedings Euromedia*, pp. 8–10, 2000.
- [24] J. Miller and J. Goodson, "A note concerning 'motion sickness' in the 2-f2-h hover trainer," tech. rep., Naval School of Aviation Medicine, 1958.
- [25] D. Johnson, "Introduction to and review of simulator sickness research," tech. rep., US Army Research Institute, April 2005.
- [26] R. S. Kennedy, N. E. Lane, K. S. Berbaum, and M. G. Lilienthal, "Simulator Sickness Questionnaire: An Enhanced Method for Quantifying Simulator Sickness," vol. 3, no. 3, pp. 203–220, 1993.
- [27] P. A. Howarth and P. J. Costello, "The occurrence of virtual simulation sickness symptoms when a HMD was used as a personal viewing system," *Displays*, vol. 18, pp. 107–116, December 1997.
- [28] L. Hettinger, K. Berbaum, and R. Kennedy, "Vection and simulator sickness," *Military Psychology*, vol. 2, no. 3, pp. 171–181, 1990.

- [29] J. Dichgans and T. Brandt, "Visual-vestibular interaction: Effects on self-motion perception and postural control," in *Perception*, pp. 755–804, Springer, 1978.
- [30] J. D. Prothero, D. E. Parker, T. Furness, and M. Wells, "Towards a robust, quantitative measure for presence," in *Proceedings of the conference on Experimental Analysis and Measurement of Situation Awareness*, pp. 359–366, 1995.
- [31] R. M. Stern, S. Hu, R. B. Anderson, H. W. Leibowitz, and K. L. Koch, "The effects of fixation and restricted visual field on vection-induced motion sickness," *Aviation, space, and environmental medicine*, 1990.
- [32] R. S. Kennedy, L. J. Hettinger, and M. G. Lilienthal, "Simulator sickness," *Motion and space sickness*, pp. 317–341, 1988.
- [33] M. J. Griffin and M. M. Newman, "Visual field effects on motion sickness in cars," *Aviation, space, and environmental medicine*, vol. 75, no. 9, pp. 739–748, 2004.
- [34] J. Irwin, "The pathology of sea-sickness," *The Lancet*, vol. 118, no. 3039, pp. 907–909, 1881.
- [35] J. Kim, C. Y. Chung, S. Nakamura, S. Palmisano, and S. K. Khuu, "The oculus rift: a cost-effective tool for studying visual-vestibular interactions in self-motion perception," *Frontiers in Psychology*, vol. 6, p. 248, 2015.
- [36] M. H. Draper, "Can your eyes make you sick? investigating the relationship between the vestibulo-ocular reflex and virtual reality," *Human Interface Technology Laboratory Technical Report*, 1996.
- [37] J. A. Sharpe and J. L. Johnston, "The vestibulo-ocular reflex: Clinical, anatomic, and physiologic correlates," *The vestibulo-ocular reflex and vertigo*, pp. 15–39, 1993.

- [38] J. Crawford and T. Vilis, "Axes of eye rotation and listing's law during rotations of the head," *Journal of Neurophysiology*, vol. 65, no. 3, pp. 407–423, 1991.
- [39] S. Tabak and H. Collewijn, "Human vestibulo-ocular responses to rapid, helmet-driven head movements," *Experimental brain research*, vol. 102, no. 2, pp. 367–378, 1994.
- [40] S. J. Herdman and R. Clendaniel, *Vestibular rehabilitation*. FA Davis, 2014.
- [41] M. Shelhamer, D. Robinson, and H. Tan, "Context-specific adaptation of the gain of the vestibulo-ocular reflex in humans.," *Journal of vestibular research: equilibrium & orientation*, 1992.
- [42] S. Reichelt, R. Häussler, G. Fütterer, and N. Leister, "Depth cues in human visual perception and their realization in 3d displays," in *SPIE Defense, Security, and Sensing*, pp. 76900B–76900B, International Society for Optics and Photonics, 2010.
- [43] S. K. Rushton and P. M. Riddell, "Developing visual systems and exposure to virtual reality and stereo displays: some concerns and speculations about the demands on accommodation and vergence," *Applied Ergonomics*, vol. 30, no. 1, pp. 69–78, 1999.
- [44] A. Sherstyuk *et al.*, "Dynamic eye convergence for head-mounted displays," in *Proceedings of the 17th ACM Symposium on Virtual Reality Software and Technology*, pp. 43–46, ACM, 2010.
- [45] T. Oskam, A. Hornung, H. Bowles, K. Mitchell, and M. H. Gross, "Oscam-optimized stereoscopic camera control for interactive 3d.," *ACM Trans. Graph.*, vol. 30, no. 6, p. 189, 2011.
- [46] R. Yao, T. Heath, A. Davies, T. Forsyth, N. Mitchell, and P. Hoberman, "Oculus VR best practices guide," tech. rep., Oculus VR, LLC, July 2014.

- [47] M. Lambooi, M. Fortuin, I. Heynderickx, and W. IJsselsteijn, "Visual discomfort and visual fatigue of stereoscopic displays: a review," *Journal of Imaging Science and Technology*, vol. 53, no. 3, pp. 30201–1, 2009.
- [48] D. Hoffman, A. Girshick, K. Akeley, and M. Banks, "Vergenceaccommodation conflicts hinder visual performance and cause visual fatigue," *Journal of Vision*, vol. 8, pp. 1–47, June 2008.
- [49] T. Shibata, J. Kim, D. Hoffman, and M. Banks, "The zone of comfort: Predicting visual discomfort with stereo displays," *Journal of Vision*, vol. 11, no. 8, pp. 1–29, 2011.
- [50] J. Kim, D. Kane, and M. S. Banks, "The rate of change of vergenceaccommodation conflict affects visual discomfort," *Vision Research*, July 2014.
- [51] S. R. Bharadwaj and T. R. Candy, "Accommodative and vergence responses to conflicting blur and disparity stimuli during development," *Journal of vision*, vol. 9, no. 11, p. 4, 2009.
- [52] T. Bando, A. Iijima, and S. Yano, "Visual fatigue caused by stereoscopic images and the search for the requirement to prevent them: A review," *Displays*, vol. 33, no. 2, pp. 76–83, 2012.
- [53] E. Viirre, "A survey of medical issues and virtual reality technology," *Virtual Reality World*, vol. 2, no. 4, p. 16, 1994.
- [54] H. T. Regenbrecht, T. W. Schubert, and F. Friedmann, "Measuring the sense of presence and its relations to fear of heights in virtual environments," *International Journal of Human-Computer Interaction*, vol. 10, no. 3, pp. 233–249, 1998.
- [55] M. E. McCauley and T. J. Sharkey, "Cybersickness: Perception of self-motion in virtual environments," *Presence: Teleoperators and Virtual Environments*, vol. 1, no. 3, pp. 311–318, 1992.

- [56] J. Moss and E. Muth, "Characteristics of head-mounted displays and their effects on simulator sickness," *Human Factors*, vol. 53, pp. 308–319, April 2011.
- [57] M. Abrash, "What VR could, should and almost certainly will be within 2 years." <http://media.steampowered.com/apps/abrashblog/Abrash%20Dev%20Days%202014.pdf>, 2014.
- [58] E. Kolasinski, "Simulator sickness in virtual environments," tech. rep., US Army Research Institute, May 1995.
- [59] M. H. Draper, E. S. Viirre, T. A. Furness, and V. J. Gawron, "Effects of image scale and system time delay on simulator sickness within head-coupled virtual environments," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 43, no. 1, pp. 129–146, 2001.
- [60] L. H. Frank, J. G. Casali, and W. W. Wierwille, "Effects of visual display and motion system delays on operator performance and uneasiness in a driving simulator," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 30, no. 2, pp. 201–217, 1988.
- [61] J. Carmack, "Latency mitigation strategies." <http://www.altdevblogaday.com/2013/02/22/latency-mitigation-strategies>. Accessed: 2015-02-05 via <http://web.archive.org> displaying content for 2013-02-25.
- [62] K. Akeley, S. J. Watt, A. R. Girshick, and M. S. Banks, "A stereo display prototype with multiple focal distances," *ACM Trans. Graph.*, vol. 23, pp. 804–813, Aug. 2004.
- [63] G. D. Love, D. M. Hoffman, P. J. Hands, J. Gao, A. K. Kirby, and M. S. Banks, "High-speed switchable lens enables the development of a volumetric stereoscopic display," *Opt. Express*, vol. 17, pp. 15716–15725, Aug 2009.

- [64] S. Liu, H. Hua, and D. Cheng, "A novel prototype for an optical see-through head-mounted display with addressable focus cues," in *IEEE transactions on visualization and computer graphics*, vol. 16, pp. 381 – 393, IEEE Computer Society, 2010.
- [65] D. Lanman and D. Luebke, "Near-eye light field displays," *ACM Trans. Graph.*, vol. 32, pp. 220:1–220:10, Nov. 2013.
- [66] J. Ehrlich and M. Singer, "Simulator sickness in stereoscopic vs. monoscopic helmet mounted displays," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, October 1996.
- [67] T. Blum, M. Wieczorek, A. Aichert, R. Tibrewal, and N. Navab, "The effect of out-of-focus blur on visual discomfort when using stereo displays," in *Mixed and Augmented Reality (ISMAR)*, 2010 9th IEEE International Symposium on, pp. 13–17, IEEE, 2010.
- [68] A. S. Percival, "The relation of convergence to accommodation and its practical bearing," *Ophthalm. Rev.*, vol. 11, pp. 313–328, 1892.
- [69] R. So, W. Wong, R. Yip, A. Lam, and P. Ting, "Benefits of matching accommodative demands to vergence demands in a binocular head-mounted display: A study on stereo fusion times," *Presence*, vol. 20, no. 6, pp. 545–558, 2011.
- [70] J. D. Prothero, M. H. Draper, T. Furness 3rd, D. E. Parker, and M. J. Wells, "The use of an independent visual background to reduce simulator side-effects.," *Aviation, space, and environmental medicine*, vol. 70, no. 3 Pt 1, pp. 277–283, 1999.
- [71] J. J.-W. Lin, H. Abi-Rached, D.-H. Kim, D. E. Parker, and T. A. Furness, "A natural independent visual background reduced simulator sickness," in *Proceedings of the human factors and ergonomics society annual meeting*, vol. 46, pp. 2124–2128, SAGE Publications, 2002.

- [72] A. T. Duchowski, D. H. House, J. Gestring, R. I. Wang, K. Krejtz, I. Krejtz, R. Mantiuk, and B. Bazyluk, "Reducing visual discomfort of 3d stereoscopic displays with gaze-contingent depth-of-field," in *Proceedings of the ACM Symposium on Applied Perception, SAP '14*, pp. 39–46, ACM, 2014.
- [73] S. Hillaire, A. Lécuyer, R. Cozot, and G. Casiez, "Using an eye-tracking system to improve camera motions and depth-of-field blur effects in virtual environments," in *Virtual Reality Conference, 2008. VR'08. IEEE*, pp. 47–50, IEEE, 2008.
- [74] M. S. B. David M Hoffman, "Focus information is used to interpret binocular images," *Journal of vision*, vol. 10, no. 5, p. 13, 2010.
- [75] H. S. SmallmanTf and D. I. MacLeod, "Spatial scale interactions in stereo sensitivity and the neural representation of binocular disparity," *Perception*, vol. 26, pp. 977–994, 1997.
- [76] S. Shiwa, K. Omura, and F. Kishino, "Proposal for a 3-d display with accommodative compensation: 3ddac," *Journal of the Society for Information Display*, vol. 4, no. 4, 1996.
- [77] F. Kooi and T. A., "Visual comfort of binocular and 3d displays," *Displays*, vol. 25, pp. 99–108, 2004.
- [78] M. Mauderer, S. Conte, M. A. Nacenta, and D. Vishwanath, "Depth perception with gaze-contingent depth of field," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '14*, (New York, NY, USA), pp. 217–226, ACM, 2014.
- [79] C. Lang, T. V. Nguyen, H. Katti, K. Yadati, M. Kankanhalli, and S. Yan, "Depth matters: Influence of depth cues on visual saliency," in *Computer Vision—ECCV 2012*, pp. 101–115, Springer, 2012.

- [80] R. Mantiuk, B. Bazyluk, and A. Tomaszewska, "Gaze-dependent depth-of-field effect rendering in virtual environments," in *Serious Games Development and Applications*, pp. 1–12, Springer, 2011.
- [81] M. Vinnikov and R. S. Allison, "Gaze-contingent depth of field in realistic scenes: The user experience," in *Proceedings of the Symposium on Eye Tracking Research and Applications*, pp. 119–126, ACM, 2014.
- [82] M. Day, D. Seidel, L. S. Gray, and N. C. Strang, "The effect of modulating ocular depth of focus upon accommodation microfluctuations in myopic and emmetropic subjects," *Vision research*, vol. 49, no. 2, pp. 211–218, 2009.
- [83] G. Kramida and A. Varshney, "Resolving the vergence-accommodation conflict in head mounted displays,"
- [84] A. Kenny, H. Koesling, and D. Delaney, "A preliminary investigation into eye gaze data in a first person shooter game," in *In Proceedings of European Conference on Modelling and Simulation, Addison-Wesley Professional*, 2005.
- [85] F. Sun, L. Stark, A. Nguyen, V. Lakshminarayanan, J. Wong, and E. Muller, "Changes in accommodation with age: Static and dynamic," vol. 65, pp. 492–498, 1988.
- [86] G. Riguer, N. Tatarchuk, and J. Isidoro, "Real-time depth of field simulation," in *ShaderX2: Shader Programming Tips & Tricks with DirectX9*, p. 539556, Plano, TX: Ed. Wordware Publishing.
- [87] R. Fernando, E. Haines, and T. Sweeney, "Gpu gems: programming techniques, tips, and tricks for real-time graphics," *Dimensions*, vol. 7, no. 4, p. 816, 2001.
- [88] P. Cozzi and C. Riccio, *OpenGL Insights*. CRC press, 2012.

- [89] T. Scheuermann *et al.*, "Advanced depth of field," *GDC 2004*, vol. 8, 2004.
- [90] H. Nguyen, *Gpu gems 3*. Addison-Wesley Professional, 2007.
- [91] Epic Games inc, "Unreal engine depth of field." <http://goo.gl/7gPydT>. Accessed: 2014-07-13.
- [92] B. Lang, "What does it look like in the oculus rift?." <http://www.roadtovr.com/what-does-it-look-like-in-the-oculus-rift/>. Accessed: 2015-02-27.
- [93] R. H. So, W. Lo, and A. T. Ho, "Effects of navigation speed on motion sickness caused by an immersive virtual environment," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 43, no. 3, pp. 452–461, 2001.
- [94] L. O'Hare, T. Zhang, H. T. Nefs, and P. B. Hibbard, "Visual discomfort and depth-of-field," *i-Perception*, vol. 4, no. 3, p. 156, 2013.
- [95] Y. Yokota, M. Aoki, K. Mizuta, Y. Ito, and N. Isu, "Motion sickness susceptibility associated with visually induced postural instability and cardiac autonomic responses in healthy subjects," *Acta otolaryngologica*, vol. 125, no. 3, pp. 280–285, 2005.
- [96] R. Bertin, C. Collet, S. Espié, and W. Graf, "Objective measurement of simulator sickness and the role of visual-vestibular conflict situations," in *Driving Simulation Conference North America*, pp. 280–293, 2005.

Appendix A

Overview of the EGD-DoF system

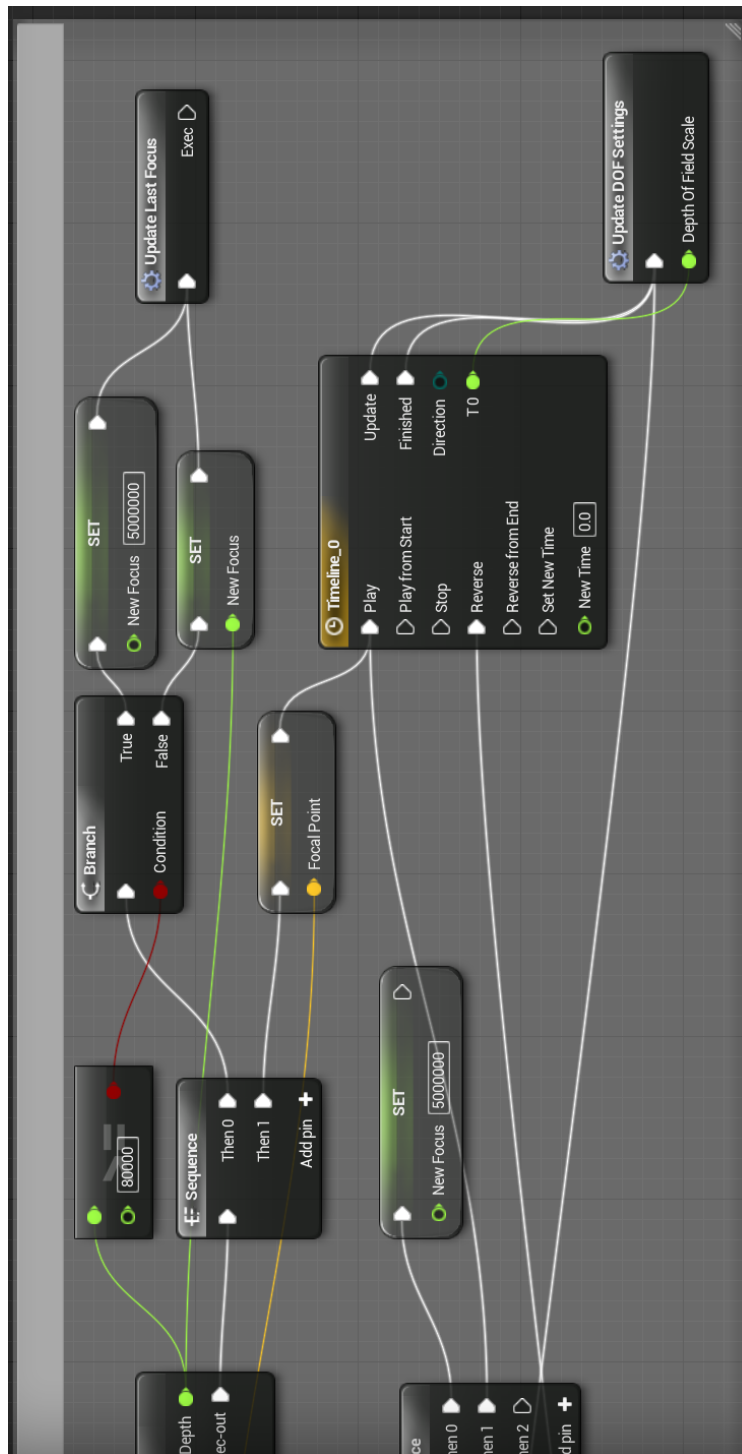


Figure A.1: UE4 Blueprint: The EGD-DoF system.

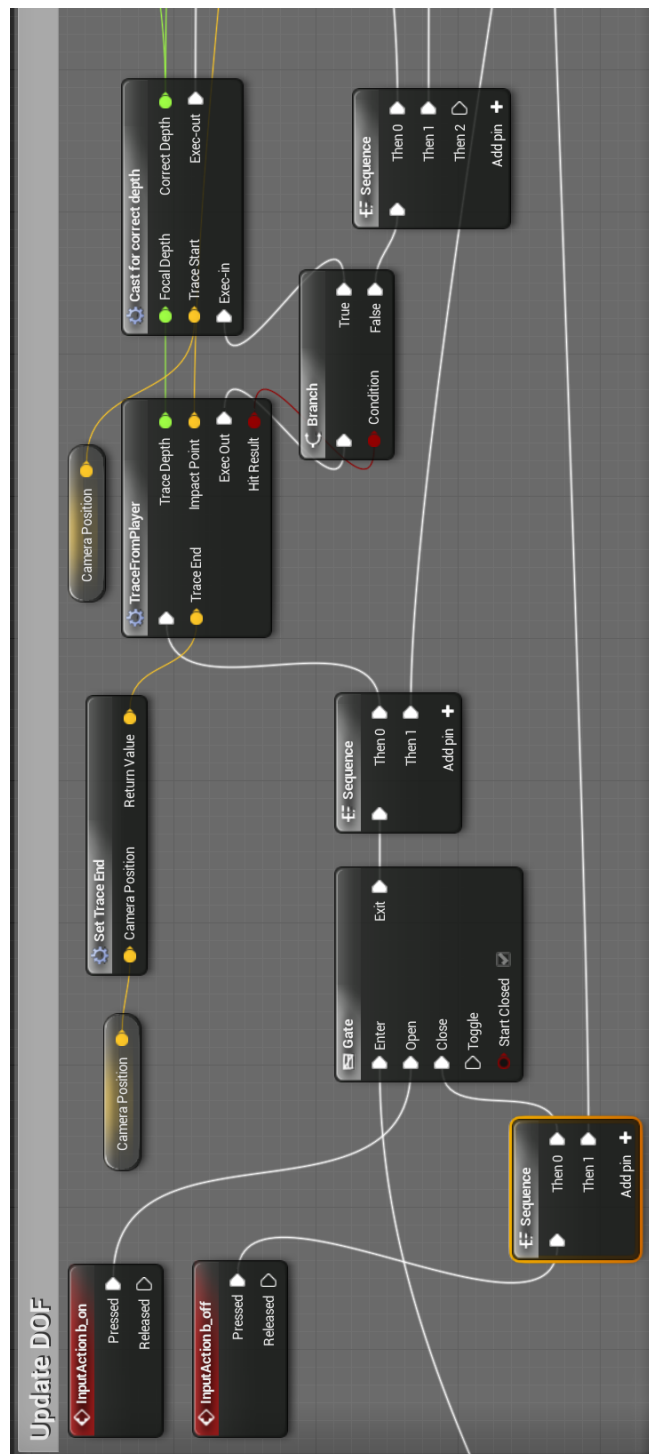


Figure A.2: UE4 Blueprint: The EGD-DoF system.

Appendix B

Renders of the scenes used in this research



Figure B.1: A high resolution screenshot of the Sun Temple scene, without the EGD-DoF system enabled

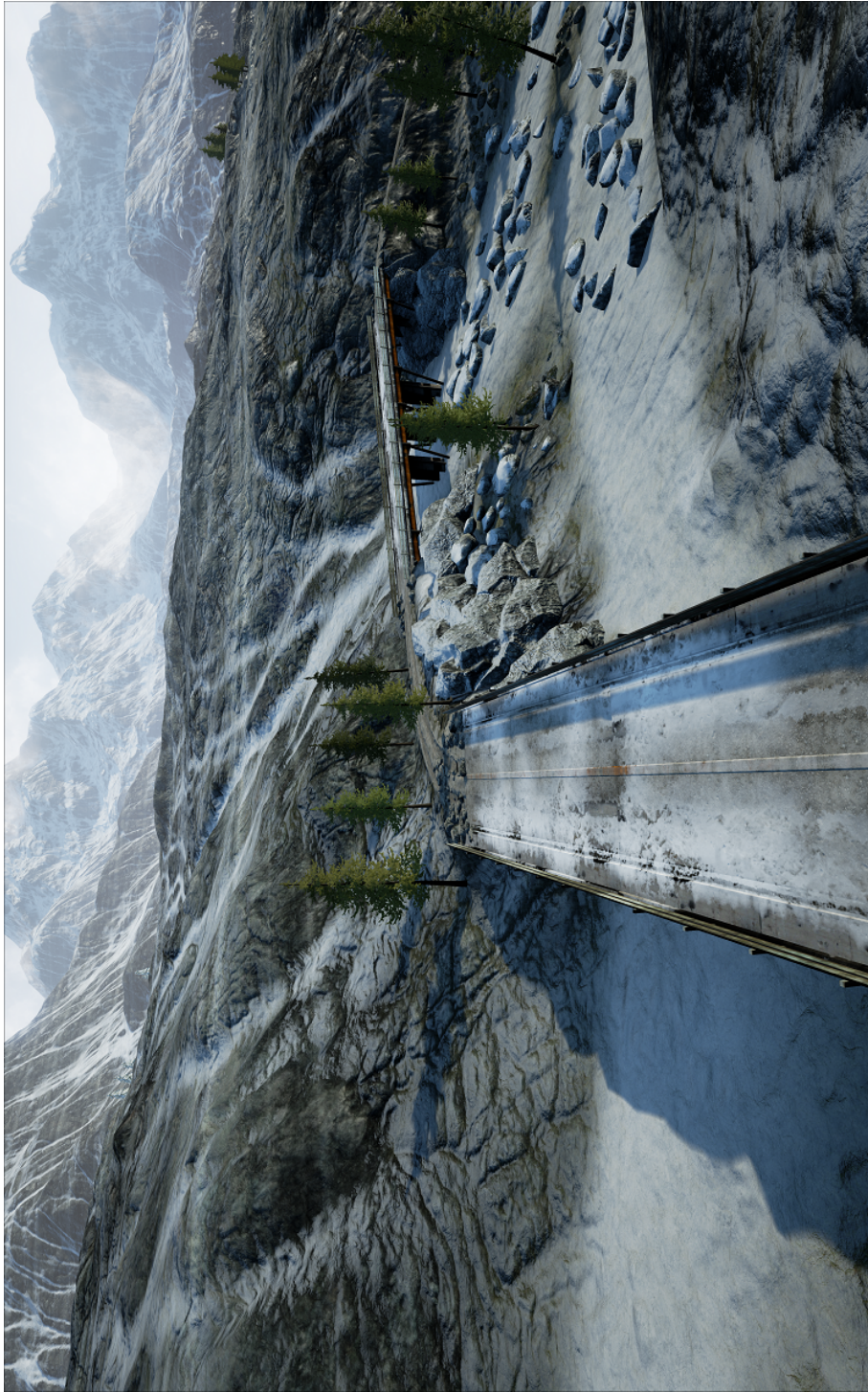


Figure B.2: A high resolution screenshot of the Landscape Mountains scene, without the EGD-DoF system enabled

Appendix C

Forms and documents used in the Study

Participate in a Computer Graphics Study

We are conducting a study involving people's level of (dis)comfort when using the Oculus Rift over a period of time. During two one hour sessions, you will explore a virtual world and answer questions on the Oculus Rift.



To participate you must:

- Not have used an Oculus rift before
- Not have any severe eye abnormalities (excluding being short/long sighted)

This research has VUW Human Ethics Committee (HEC) approval.

For more information please either phone me at **+04-463 5233, extn 8286**

OR e-mail me kotarou@ecs.vuw.ac.nz.

[illegible]



Investigating Simulator Discomfort in Current Generation Head Mounted Displays

Kieran Carnegie; Masters research project; Victoria University of Wellington

Phase 1 Participant Information sheet

You are invited to take part in a study on *Simulator Discomfort on Current Generation Head Mounted Displays (HMDs)*. This involves an interview in which you will be asked to complete a series of tasks while using either a typical 3D LCD computer screen or a Oculus Rift HMD. You will then be asked a series of questions about your experience on the device.

This study will take a maximum of one hour. Data collected during this study will be stored in an anonymous form, in which no personally identifying information about you will be kept. You may withdraw from this study at any time, both prior to the interview commencing, and during the interview, should you feel it necessary.

Should you wish to withdraw from the study after the interview, please contact me before 1 Oct, 2014 to have your results removed.

This Participant Information Sheet will help you decide if you'd like to take part. It sets out why I am doing the study, what your participation would involve, what the benefits and risks to you might be, and what would happen after the study ends. Feel free to ask me any questions.

If you agree to take part in this study, you will be asked to sign the Consent Form on the last page of this document.

Please make sure you have read and understood both this Information Sheet and the Consent Form before signing.

This research has VUW Human Ethics Committee (HEC) approval.

What are the conditions for my participation in this study?

In order to participate in this study, you cannot meet any of the following criteria:

1. Have any prior experience on a HMD
2. Be pregnant
3. Have a severe predisposition to motion sickness
4. Have undergone eye surgery within the last year
5. Have any eye abnormalities* apart from the following exceptions:
 - a. Short/long sightedness
 - b. Heterochromia and other cosmetic changes
 - c. Colour Blindness

* Eye abnormalities include but are not limited to: Lazy eyes, spherical/chromatic aberrations...

If none of the above apply to you, then you are eligible for participation in this study.

What is the purpose of this Study?

In this study, I aim to better understand what factors in Head Mounted Displays lead to "Simulator Discomfort", such that I can develop a solution that will reduce the incidence of discomfort people undergo when using HMDs like the Oculus Rift.

What will participation in this study involve?

You will be asked to either use a 3DLCD screen or a HMD, and complete a set of virtual tasks, which will take around 30min.

At the beginning of the stud, once you complete the virtual tasks, and 15 minutes after completing them, you will be asked a series of questions relating to your current comfort levels. Example questions include:

1. On a scale of 1-5, how much of a headache do you currently have?
2. On a scale of 1-5, how would you describe the dryness of your eyes?

Again: at any point, you are free to terminate the study. I will ask for a reason, but you do not have to provide one.

What are the possible benefits and/or risks of this study?

It is possible that you will develop one or several of the following symptoms during this study: headache, nausea, double vision, dry / burning / irritated eyes, fatigue and dizziness.

These symptoms recede within a short period of time after HMD use, typically within 10 or so minutes.

What happens once this study ends?

I will remove all identifying information about you from the results, and only store aggregated data: you may request from me these results, but they will not be tailored to you or mention you in any specific manner.

These aggregate results will be stored indefinitely.

The aggregates results of this study will used in my masters thesis, and in a proposed research paper to be published this year.

If you are interested in finding out more and participating in this study please either phone me at **+04-463 5233, extn 8286** OR e-mail me kotarou@ecs.vuw.ac.nz.

Thank you very much for your time and help in making this study possible.



Investigating Simulator Discomfort in Current Generation Head Mounted Displays

Kieran Carnegie; Masters research project; Victoria University of Wellington

Participant Consent form

Please read the following notices. If you agree with them, please sign this form.

By signing this sheet: I, the participant, understand and agree that:

- *My participation in this research is voluntary and I am aware that I am able to withdraw at any time, including after my interview, before the 1st of October, 2014.*
- *I am aware that, should I choose to withdraw, the investigator may ask me why I have chosen to withdraw but I am not compelled to answer if I do want to.*
- *I confirm that I have have been provided, read and understand the Participant Information Sheet.*
- *I have had the opportunity to ask any questions about this research and had them answered.*
- *I understand that all personal information will remain confidential and that all efforts will be made to ensure I cannot be identified (except as might be required by law).*
- *I agree that data gathered in this study may be stored anonymously and securely, and may be used for future research.*
- *I agree to take part in this study.*
- *If I have any further concerns and/or questions, I am aware I can contact the investigator at: +04-463 5233, extn 8286 or via email at: kotarou@vuw.ac.nz*

I wish to receive a copy of the results of this study: Yes No

Participant's name & signature:

Investigator's signature:

Date:



Investigating Simulator Discomfort in Current Generation Head Mounted Displays

Kieran Carnegie; Masters research project; Victoria University of Wellington

Phase 1 Discomfort Questionnaire

Participant Name:

SYMPTOM CHECKLIST

Please fill in this questionnaire. On a scale of 1 to 5, with 1 meaning "No symptoms" and 5 meaning "Severe Symptoms", how would you rate your current level of:

1. General discomfort	1	2	3	4	5
2. Fatigue	1	2	3	4	5
3. Boredom	1	2	3	4	5
4. Drowsiness	1	2	3	4	5
5. Headache	1	2	3	4	5
6. Eyestrain	1	2	3	4	5
7. Difficulty focusing	1	2	3	4	5
8. Blurred vision	1	2	3	4	5
9. Salivation increase	1	2	3	4	5
10. Dry Mouth	1	2	3	4	5
11. Sweating	1	2	3	4	5
12. Nausea	1	2	3	4	5
13. Difficulty concentrating 1	2	3	4	5	
14. "Fullness of the head"	1	2	3	4	5
15. Anxiety	1	2	3	4	5
16. Dizziness	1	2	3	4	5
17. Vertigo	1	2	3	4	5
18. Visual flashbacks	1	2	3	4	5

19. Aware of breathing	1	2	3	4	5
20. Stomach awareness	1	2	3	4	5
21. Loss of appetite	1	2	3	4	5
22. Desire to move bowels	1	2	3	4	5
23. Confusion	1	2	3	4	5

This document is a modified version of the questionnaire available here:
http://www.cybersickness.org/Simulator_Sickness_Questionnaire.html

Investigator use only:

Participant completed:	Yes		No		NA
Stage Asked:	1	2	3	4	

Notes: