UNIVERSITY OF CAPE TOWN

DEPARTMENT OF COMPUTER SCIENCE

CS/IT Honours
Final Paper 2020

Title: The Effect of Interaction on Eliciting Fear in Virtual Reality

Author: Jordan Taschner

Project Abbreviation: VREmote

Supervisor(s): James Gain, Gosia Lipinska, Siphumelele Sigwebela

Category	*Min*	*Max*	Chosen
Requirement Analysis and Design	*0*	*20*	15
Theoretical Analysis	*0*	*25*	0
Experiment Design and Execution	*0*	*20*	5
System Development and Implementation	*0*	*20*	20
Results, Findings and Conclusions	*10*	*20*	10
Aim Formulation and Background Work	*10*	*15*	10
Quality of Paper Writing and Presentation	*10*		10
Quality of Deliverables	*10*		10
Overall General Project Evaluation (*this section allowed only with motivation letter from supervisor*)	*0*	*10*	0
Total marks		**80**	80

The Effect of Interaction on Eliciting Fear in Virtual Reality

Jordan Taschner
TSCJOR001@myuct.ac.za
Department of Computer Science
University of Cape Town

ABSTRACT

Exposure therapy is a technique in psychology that exposes the patient to the source of their anxiety in a controlled environment, in hopes that the patient overcomes their distress. [1]. Emotion elicitation in Virtual Environments (VEs) serves as a novel approach for contemporary exposure therapy. These environments provide an entirely controlled medium to the patient and can evoke almost any emotion. Extant VEs have been successful in this area of treatment [5, 6], but most rely on traditional methods of eliciting emotion, these being, visual, audio and haptic feedback triggers placed in the environment that are experienced in a linear and guided fashion. We developed a new iteration of an extant VE that introduces interaction in conjunction with Artificial Intelligence (AI). We aim to determine if a novel interactive approach can elicit fear in a controlled, measured and reliable way, far beyond traditional methods, by comparing these new interactive techniques against the already present triggers. To accurately define these comparisons, we created a three-tier interaction system that varies the VE from no-to-moderate-to-high levels of interaction, and conducted an evaluation with three experts in the field. The evaluators experienced all three tiers of interaction, and completed two forms (VR heuristic evaluation and open form emotion questionnaire) to develop a proper indication that any form of interaction can objectively elicit more fear over no interaction. This form of evaluation was chosen due to the onset of COVID-19, barring intended user testing – which requires analysis of the autonomic nervous system. Results indicate that interaction and AI can elicit higher levels of fear over that of an environment without interaction, but further, more conclusive tests need to occur, as only three evaluations lack enough objective evidence. Usability heuristics justifies the environment use of this environment for these future studies, without major development changes.

1 INTRODUCTION

Jonathan Steuer of Stanford [2] refers to VR as a medium that is not only defined through the product. Head-Mounted Displays (HMDs), motion gloves and other virtual world technologies only provide the definition of VR from a marketing standpoint, a standpoint he says was coined by Krueger in 1991, a leader in VR technology production at the time. Steuer rather suggests that VR is something more psychophysiological; a sense of presence – a feeling of being *really there* in a world other than reality. This definition accurately defines contemporary VR. The medium transcends that of a simple lens connected to high resolution displays. VR is something more capable, something that can trick us and elude us into experiences that can only be dreamt of. This more abstract definition of VR developed the foundations of its use in the field of psychology. By developing and utilizing carefully controlled virtual worlds, psychologists began investigation into the clinical applications of VR. These virtual environments could be therapeutic outlets to users who suffer from depression, anxiety and other mental illnesses, e.g. PTSD treatment [3]. It was then found that this treatment could be similarly used in exposure therapy, a technique in behaviour therapy to treat phobias and anxiety disorders by exposing the patient to a controlled manifestation of the source of their anxiety [5, 6]. By developing a controlled virtual environment that elicits specific emotions, one could treat that specific emotional deficit in a patient.

While it is evident that exposure therapy using VR can successfully treat anxiety and phobias, most research now lies in development of environments that elicit targeted emotions in the most efficient and controlled way possible. Often the environments created in these studies guide the user through a predictable virtual world, and expose them to various scripted audio, visual and haptic triggers (or cues) designed to elicit specific emotions. In a fear VE, these non-interactive triggers can execute stimuli such as a jump-scare when opening a door [4]. However, contrary to the intention of a controlled environment, the user has no control over whether they wish to experience such triggers. The user in the case of jump-scare cannot choose not to experience it, and while it can be effective in eliciting fear, the excessive use of these scripted triggers can create redundancy, expectation, or a response that elicits too much fear. The use of scripted triggers has now become a saturated field of research, and has been conclusively determined as an effective method of eliciting fear. What is interesting is the lack of research in methods other than scripted triggers, the most notable being interaction and believable AI in the elicitation of fear, and if such applications could manipulate levels of fear in an innovative way, we could fine-tune fear reactions for therapeutic outcomes, while maintaining a sense of user control.

We propose a virtual environment that can create both a higher level of fear over the traditional use of scripted triggers, while maintaining a sense of agency for the user. We also discuss the effect of manipulation of this agency and how it can control the amount of fear the user experiences – by creating three different levels of interaction in the virtual environment, each varying the amount of interaction and AI the user can experience. Our project is a new iteration of an environment that was previously developed. The previous iteration contained most of the assets that were used for our new environment, and only made use of scripted triggers to induce fear. Details of the previous iteration can be found in subsection 4.4.

Due to the onset of COVID-19 at the beginning of development, original intentions of user-testing of the autonomic nervous system where set aside for future iterations our project. Instead, we opted for three expert evaluators to determine the development status of the project in preparation for future studies, and also if interaction can, in their opinion, elicit higher levels of fear. Performance metrics such as framerate and user comfortability in the VE were also analyzed.

This leads us to the aims of our project:

Aim: Develop a VE that focuses on user control and interaction, and allow the user to choose between three varying levels of interaction.
Aim: Develop believable AI in the VE that aids in interaction in creating levels of fear above that of traditional methods, such as scripted triggers.

And our research question:

Aim: To determine whether interaction can be used as a novelty for eliciting fear, and if this environment is fit for future study.
Hypothesis: The environment does elicit higher levels of interaction according to our expert evaluators. Heuristic evaluation presents problems that are within our reach to fix, and performance of the environment is optimal, enabling the foundations for future studies with this environment.

This report consists of 8 sections. Section 2 is an overview of related work that enabled proper execution of our development. Section 3 focuses on our design process and tools. Section 4 highlights the virtual environment and its ability to induce fear before our additions, and section 5 outlines our interaction and AI after additions. Section 6 and 7 details our evaluations and results, and section 8 concludes our project.

2 RELATED WORK

2.1 Defining Immersion

Emotion in VR has been extensively studied in the past, and usually refers to the concepts of immersion and presence as drivers that correlate to the elicitation of emotions. It is often found that an increase immersion or presence results in a direct increase in emotional response [7, 8, 9]. Immersion in a VR refers to the technology behind the experience (the headset, motion controllers, realistic tracking etc.) and represents the objective experience of the user [10]. That is the technology that creates the illusion of being in a real environment. Presence is the subjective part of the experience, or when the user feels as if they are *really there*. This is related to the users personal engagement with the virtual world [10]. To avoid confusion, we refer to immersion and presence as simply immersion by using Mutterleins definition [11]: immersion is a subjective psychological experience which is restricted by the technology that enables that experience. We can use this definition to assume that if studies suggest that presence or immersion or a combination of both evoke an emotion, immersion does so equally.

2.2 The Psychology of Fear

To understand why virtual environments that induce fear are successful in doing so, an underlying understanding of why our minds react to fear stimuli in certain ways is required. The psychology of how fear is rooted in our brains, and why we feel the emotion of fear, is influenced by the development of evolutionary psychology and past trauma.

2.2.1 Evolutionary Psychology. The evolutionary development of fear relates to the innate struggle of the "survival of the fittest" throughout history. Natural selection also plays a key role here – survival when faced with instances of encroaching harm are carried through the genes of our ancestors. Fear as an emotion is defined through the results of these challenges - or adaptions [12], and if such adaptions are successful, their positive and negative characteristics are retained in each subsequent generation [13]. The negative characteristics are those that are vital to the essence of fear as an emotion. Cosmides and Tooby [12] provide an example to illustrate this concept. When humans are alone at night, we perceive the presence of other humans or predators and feel as if we are being watched, as a past adaption involved increased violence during the night. This cues a behavioural response, a heightened sense of attention to the surrounding environment, and a need to protect one's self. In a virtual environment, this scenario can be used in conjunction with a very dark setting, presenting the user as completely alone, and making the user aware that threats are in fact present. When the user encounters the threat, the user usually can use something to defend themselves [14] or simply avoid the threat completely (*Alien: Isolation*, The Creative Assembly, 2014). Virtual environments can use this technique by creating threatening situations or placing the user in isolated environments, and implementing abrupt noises and visual cues such as a door slam. This implies the presence of something or someone else causing that cue, and can instinctively create a need to survive [4].

2.2.2 Trauma. Past trauma and PTSD are also important factors to consider when subjecting the user to fearful virtual environments. It is vital to both the results of an emotion elicitation experiment, and to the participant that the user is clear of any significant past or present mental stressors. Exposing the user to an event that can relate to their stressor can result in additional psychological trauma and skewed results [15], and is imperative to avoid.

2.3 Fear Virtual Environments

2.3.1 Realism and Hardware. Immersion has a very close correlation to realism. Realism in a virtual environment can relate to the hardware used for the experience, or high graphical fidelity within the environment. Both of these aspects represent an objective view on immersion. A user who, for example, experiences an environment with an Oculus Development Kit I (2012) VR headset, and then experiences the same environment on a HTC Vive (2016), would testify that the Vive provided a conclusively higher sense of immersion. This is because of the hardware generation gap: the Vive simply has a higher FOV, higher refresh rate, and a higher overall resolution of the display panel. Abrash [16] notes that all three of these properties, FOV, refresh rate and resolution are key factors in providing higher levels of immersion for the average user. A wide FOV, provided by the lens and display of the HMD, enables the user to see more of the virtual world, and thus develops a higher sense of plausibility illusion [17] – the illusion that a scene is actually occurring [18]. A higher refresh rate can create smoother motion, as more frames are updated per second on the display. A smoother motion reduces motion sickness; which can pull users completely out of the experience due to nausea [19]. Finally a denser resolution results in a clearer image, which presents virtual objects as sharper and closer to reality. Since the HMD is pressed against the users face, individual pixels can be seen, distracting users from the experience, causing a decrease in immersion. But a clearer image can only provide clarity on the objects of the virtual world, meaning these objects need to enable realism themselves. Objects with higher polygon counts and higher texture resolution have shown to increase realism (see figure 1), and are imperative to maintaining immersion, especially in a fear environment [7]. However, too much realism and the uncanny valley effect can occur [20], whereby users are presented with a humanoid model that tries to closely resemble that of the real world, evoking a sense of eeriness or even fear in the user. While this does not apply to non-humanoid models, a distracting effect from models that try to resemble normal objects could also occur.

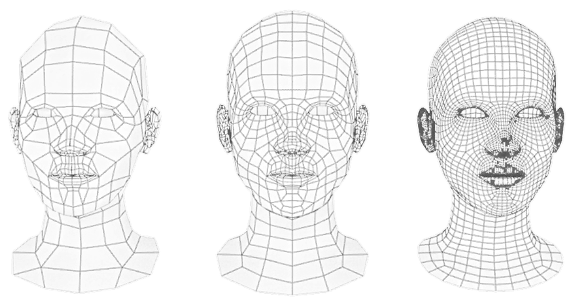


Figure 1: Realism - low polygon vs. high polygon models

2.3.2 Atmosphere and Scripted Cues. Fear as an emotion inherently relies on the ability of a virtual environment to produce an eerie or creepy effect on the user. Environments that are dark with low visibility, resembling a picturesque resemblance to a naturally scary area, such as a graveyard or abandoned house, and utilize sound and scripted triggers create an overall epitome of a fear-inducing environment. These environments must be carefully manufactured in terms of timing of fear exposure, and how controlled such exposure is. [9], an environment that places the user in a dark apartment, found strong evidence that atmospheric environments composed of audio and visual cues such as door slams and television static evoked a strong sense of immersion and fear response in the user. [21], a guided experience through a dark corridor, found that a culmination of impending harm through scripted visual frights create good fear response. [22], another guided experience where the user is subjected to a T-rex attack, found that haptic feedback and visual cues also increased a fear reaction. All of these environments have one key indicator in common: there is always a fear of the unknown, and whether that is created through scripted cues or dark settings, they are powerful elicitors of fear.

2.3.3 Interaction and Agency. Often the VEs that are designed in past studies find evidence that immersion is increased and thus fear response also increased when the user is subjected to scripted audio and visual cues [21, 22]. The use of these scripted cues, especially in fear evocation, is now saturated, and few studies refer to any novel techniques that make use of interaction as a technique to elicit emotion. However, a study that made use of an interactive environment showed promise in evoking emotion above that of an environment that uses traditional scripted and audio and visual cues [14]. In this environment, there is a direct manipulation of user agency, which results in a comparatively higher response in fear than visual and audio cues. The user must fight off waves of zombies in an abandoned house by interacting with a pistol; but as soon as the user is visibly revoked of using their weapon (by the program breaking the weapon), the tension and inevitable attack of the zombies create unprecedented levels of fear as they try to fix it. Agency is important here, lack of freedom that was previously available manipulates the user's control, the only aspect that was keeping them calm. Applying this concept to a more varied environment, and an array of possibilities arise. An environment that finds a balance of control and autonomy, can give the user freedom of choice while the environment can manipulate their choices, creating confusion, tension and ultimately fear. A simple addition to an environment, such as virtual hands that can interact with a variety of objects that drive the experience, can be extremely effective. Or adding a stalking monster with believable AI that can be scared off using an object, but that object could be broken or taken away, can create a menagerie of effects on the user. Studies of manipulation of agency using interaction in environments are lacking, and investigation of this field of research can be very promising. Which is evidence of why this is primary focus of this study.

3 SYSTEM DEVELOPMENT AND DESIGN

3.1 Development Tools

The entirety of the project was developed in VR, with VR in mind. This has numerous implications, from the perspective of development tools, languages, performance and user health. Throughout development, there was emphasis on understanding the user experience (UX) more than the user interface, and careful attention was paid towards usability, accessibility and affordance: from menu interaction to interaction with the game world itself.

The project was developed in the Unity Game Engine (Unity Technologies) in C#. Unity uses GameObjects to represent the objects in the virtual world, and can be parented and childed according to the use case of other objects. If an object is childed, the objects position and rotation – call the *transform* - inherits the parent GameObject's transform. For example, if the players body is the parent, and the game camera is the child, the game camera will follow the movement and rotation of the players body. GameObjects are comprised of "components," which can be anything from colliders to textures to UI elements.

For VR, Unity uses the OpenVR SDK (software development kit) that allows use of a headset. We are using the HTC Vive for development and testing. The Vive is a modern VR headset comprised of two OLED panels each with a display that has a resolution of 1080 x 1200 (combined 2160 x 1200) pixels, a 90Hz refresh rate and a 110 degree field of view (FOV). This provides optimal immersion and a strong perception of presence in the virtual world [23]. This, however, creates a performance trade-off. Providing a high framerate (>90) for high-resolution rendering requires a powerful GPU in the users' computer. A GPU, such as an Nvidia GTX 1070 (or AMD equivalent RX 590), is essential, as performance in VR is especially vital, and is a priority for optimal user comfort. Stutter from performance issues can induce simulator sickness (SS) and motion discomfort, a disorienting effect that can render the user unable to continue using the headset [24]. With this in mind, we aimed to develop our VE in a way that performs optimally, minimizing high polygon assets, culling, and reducing distance rendering of unnecessary objects. In a fear setting, this was not too hard to maintain, as we could use darkness in the environment to mask objects that have lower graphical fidelity and that are far away from the user's position. Realism, however, is to be considered when optimizing performance – as realism is achieved through realistic models – which in turn usually have higher polygon counts. It is also found to have a direct impact on evoking fear [7], so balancing sufficient realism with adequate performance was a difficult task. Too much realism can also result in the "Uncanny Valley" effect [20], distracting the user from the experience. These aspects of the VE, motion sickness and the uncanny valley, were carefully considered in terms of user comfort and avoided throughout development.

Since this project focuses primarily on interaction that closely resembles using hands to interact with GameObjects, virtual hand models were required. To create functionality for these hands, the same OpenVR SDK with support for the Vive Motion Controllers was used. The Vive Motion Controller consists of a touchpad accessible via your thumb, two grip buttons, two circular buttons and a trigger that is accessible by your index finger (see figure 2). They register movement and rotation in real time, by utilizing the Vive base stations for infrared tracking. This means that user must experience the environment in room-scale, with a play area of at least 2 x 1.5m [25], providing space for movement and using the hands without obstruction. The legacy Unity XR input system was chosen to map the inputs from the controller to Unity's input array. For our project, only the trigger, touchpad and menu buttons were mapped for use of interacting the virtual world. The touchpad and trigger were mapped as axes, allowing us to receive intermediate float values depending on how the input was received. For example, if the user holds the trigger down halfway, a float value of 0.5f would be returned. This is useful for precise input mapping and registering specific actions made by the user. The menu button was mapped as a button, and was used simply to determine if the user had pressed the button down.

3.2 Design Methodology

In order to maintain and create a workflow that optimized user feedback and content delivery, the User-Centred Design (UCD) paradigm for developing software was adopted. UCD revolves around iterative development, where product releases are reviewed and then redesigned following user feedback. This cycle continues until the final product is released, and if done correctly, results in a professional yet accessible piece of work – as the user is the primary stakeholder in experiencing the product [26]. Every 10 days, we would convene with our supervisors in video meetings, and reflect on all previous work done. Our supervisors would then suggest a variety of changes and we would assess viability. Our supervisors consisted of experts in the field of Computer Science and Psychology, with an additional student of psychology adding their thoughts in each meeting. Once a new iteration was agreed upon, development began immediately. This cycle continued for numerous months leading to the final build of the fear virtual environment.

This methodology was adopted as the proposed changes to the previous virtual environment - our inclusion of interaction and AI - were experimental. Our UCD development consisted of three crucial stages among our consistent meeting schedule. The first was a Games Design Document (GDD) that outlined world design and mechanics of our new environment. This included a full proposal and design of AI and Interaction additions and changes. This document was sent in month one to our supervisors for review. Following this, our additions were assessed, and confirmed for the start of development. The next two stages were the showcase of each demo. The first demo outlined the success of the foundations of developing our new additions, and the second demo provided a full look at an almost completed environment; where only small changes were to be made, leading up to the final build. Both demos acted as a visual representation of our progress, and provided us with new insight on moderate changes to be made for each new iteration of the VE.

Figure 2: HTC Vive Motion Controller

These demos were essentially prototypes. The first was low-fidelity, as we purposely remained open to criticism, hoping to make substantial changes based on supervisor feedback. It also included paper prototypes of the fundamental map design of the environment and the user's experience. The second was high-fidelity and resembled closely what was going to be the final build of the fear VE. Only minor changes could be made following this demo, as time was constrained.

This design process was ultimately successful in our development process and resulted in an objectively polished build that was ready for user experiments.

4 THE VIRTUAL ENVIRONMENT

The virtual environment resembles a damp, dark and empty sewer, comprised of various canal troughs filled with water. The user is placed in a boat that sits in the middle of a canal. The experience ends when the user reaches the end of the sewers; but while doing so, the user is being stalked by a monster that lurks around the map, intending to evoke fear. This is not a video game, as there is no objective but to simply be exposed to a controlled environment in which the users fear is manipulated for therapeutic purposes. The environment makes use of scripted AV triggers, interaction and AI as methods of eliciting fear.

4.1 Setting and Atmosphere

The experience begins at the origin (x = 0, y = 0, z = 0) of the VE (see figures 3 and 4), where the user is placed in a moderately-sized rowing boat. This boat is placed in the relative centre of a canal

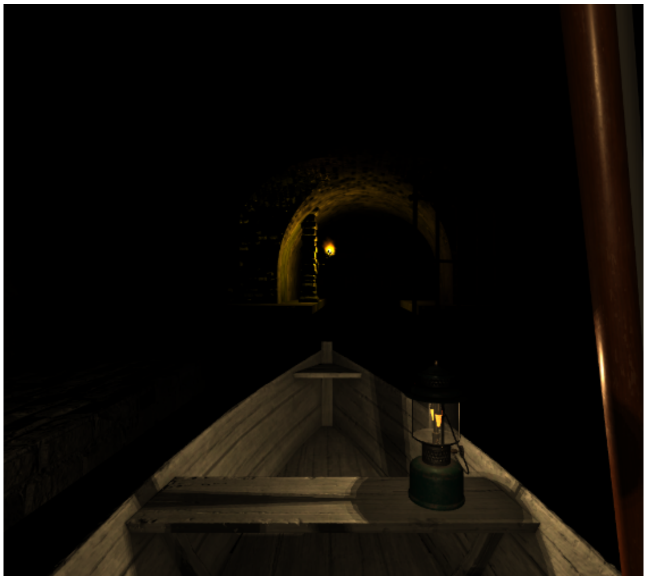


Figure 3: Starting view (directional lighting off)

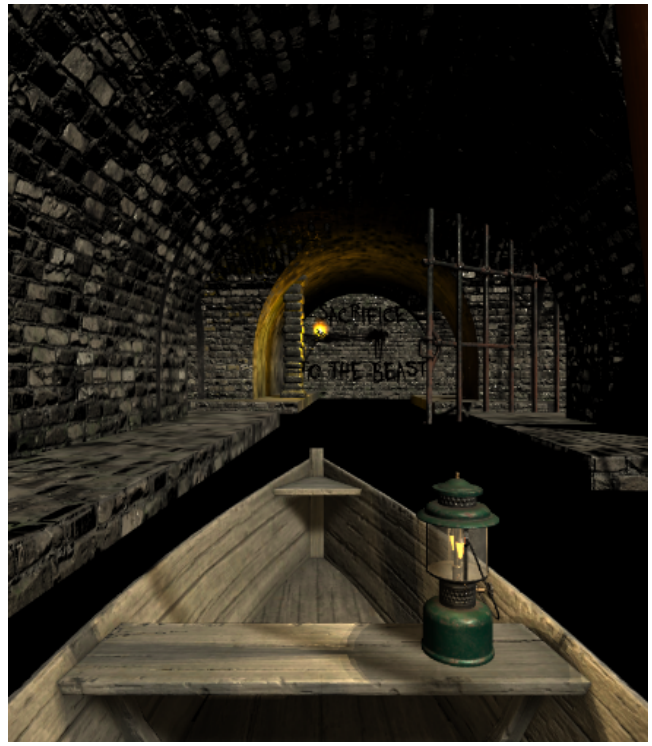


Figure 4: Starting view (directional lighting on)

trough that is filled with water. The water GameObject is an elliptical body of water that covers the trough and is offset just below the edges of the canal. Since directional lighting is disabled and user cannot see too much detail, the water object uses a simple, low resolution reflective shader that captures adequate reflections of objects, while maintaining high performance. The canal and water continues forward into darkness beyond the users field of view (FOV), and is surrounded by a shell of textured brick walls that resemble a sewer. The materials of the mesh renderer are modified to enhance specular lighting, creating a reflective effect that composes the sewer with the illusion of dampness; that the walls are wet. While the resolution of these texture maps are high, performance is enhanced by culling – or *not rendering* - parts of the walls that are not within player-view, that is, only the inner shell of the canals are rendered by the game camera. The combination of the walls and canals create a parented GameObject called "Canal Segment", and each segment has four varied types. These types dictate whether the segment is a four-way intersection of canals, or a three-way split, and so forth. The entire environment is created using these segments, and in conjunction create a highly realistic sewer environment. Other GameObjects and assets that contribute to the setting of the environment include various metal gates, bridges, pipes and their broken counterparts. These objects are lined along the canals, which the user can notice throughout their experience.

Darkness and the fear of the unknown is imperative to inducing the mood and tone of the experience, and while the assets of the sewers gives an impression of an uncomfortable and eerie setting to the user, the atmosphere, composed of lighting and sound, is meant to produce the inherent feeling of fear. Directional lighting is disabled throughout the experience, which creates a reliance on only the GameObjects in the world to create light. In VR, there is absolute darkness; as the skybox – the background of the virtual world - of the VE is set to matte black and the user can only see through the headset lens. The wide FOV of the HTC Vive further increases this notion of the surrounding absolute darkness and induces a sense of isolation and loneliness. The only light fixtures that are present, are fire torches that emit orange and yellow area light, illuminating the specular walls of the sewers. This is enough light to allow the user to catch glimpses of the monster that lurks in the background, or the broken pipes and gates that line the canals. These lights were also scripted to flicker upon the users approach, creating a signature fear trope most users can associate with a *creepy* atmosphere. Ambient sound is also a large focus to further develop the mood of the VE. An echoing, wind and tunnel soundscape is used as the primary ambient noise that the user can hear throughout their experience. This develops an immersive and suspenseful sound, and, interspersed with occasional sounds of water flowing and dripping, creates an expectation of an impending scare [27]. The environment utilizes 3D binaural audio sources for individual audio clips, to create an effect of distance and spatial sound. These sounds are placed in certain areas of the VE, usually around the corners of canals. Sound clips include the echo of a woman screaming, the snarl of the monster, water dripping from pipes and the creaking of the boat and metal gates. When the user approaches these audio sources, the sound becomes more apparent, and in combination with the darkness, creates a sense of uneasiness in the user.

4.2 The Monster

The monster in the VE is the main source of fear and is intended to scare the user the most. The monster is portrayed to the user through various visual and audio triggers throughout the experience and resembles that of a "Demogorgon" from the pop-culture hit *Stranger Things* [28]. The asset is a complicated rig of 98 bones and high resolution (4096 x 4096) textures, with 25 professional animation sequences and various audio clips [29]. Four instances of the monster are placed around the environment to perform various tasks. Each instance looks identical to give the illusion of that the same monster is lurking near the user.

4.3 Scripted Triggers and Objects

The core focus of the scripted triggers in the environment revolve around the monster and the monster's "stalking" behaviour. These triggers are executed once the player reaches certain box colliders, which are placed at strategic points for optimal effect. The box colliders only react when on object of the "player" tag registers a collision – i.e. the boat - and then executes a coroutine script that portrays the audio or visual cue. The use of coroutines is used heavily here; as coroutine events in Unity allow asynchronous method calls that can yield time delays with precision. The "falling pipe" visual trigger is executed upon user collision, and executes coroutines that wait a few seconds before calling various methods in a linear fashion. The broken pipe on the ceiling of the sewer changes state from hanging off a piece of the pipe, to breaking off into the water in the canal below, to slowly sinking when the player reaches the pipe. The monster triggers use coroutine events in a similar way. The intention of the monster is to induce fear and not to harm the user in the sense that no monster triggers end the experience early as a result of their execution. These monster triggers are executed by use of colliders, just as the broken pipe trigger did so. The "Monster Gate Crash" is triggered on a canal corner, where the user can see a gate being charged and broken by the monster. The gate is illuminated by use of one of the industrial light fixtures that flickers on arrival. "Gate Close Behind Player" is then executed immediately after the player goes through the same broken gate, which shuts close abruptly, intending to create a fright. Similarly, "Monster Catwalk" depicts the monster running across a metal bridge in front of the player. Finally, "Monster Attack" is a trigger that is executed at the end of the players guided path. The monster runs towards the user who is forced to subsequently watch the monster attack the user by clawing and roaring at them, executing associated animations and sounds. Once the animations are complete, the experience ends by fading the screen to black.
Non-visual triggers are purely sound triggers. Some of these are executed when the monster performs the visual cues, others are placed with colliders around the environment. Examples include ambient monster roaring, people screaming, creaking of the gates and pipes and knocking on the walls. All make use of the same binaural 3D sound technique for enhanced audio fidelity, and are parented under the GameObject "SoundFX." For the full list of visual and audio triggers, see supplementary figure S2.

4.4 Previous Environment Work

The previous environment provided us with most of the core assets for our new environment. The previous VE assets were all GameObjects mentioned in the above section, including the canal assets, ambient sound effects, water, the monster and the scripted triggers. Using these assets, the previous environment was ultimately successful in eliciting higher levels of emotional arousal, with sympathetic nervous system metrics in favour of fear evocation (faster heartbeat, higher skin conductance response) when experiencing the scripted triggers [30]. The intuitive emotional response would be higher levels of surprise and fear; both of which were also satisfied in the environment. In our new iteration of the VE, we opted to retain assets that were the sources of fear elicitation – the setting, monster and scripted triggers – to get a fair comparison when contrasted with our additions of interaction and AI. We also retained the initial map layout and audio, which was expanded on further in development (see supplementary figure S5).

With context now in mind, we began to significantly change and adapt the environment to utilize interaction and AI.

5 INTERACTION AND AI

5.1 Interaction

The core additions of the new environment revolve around interaction and AI and their ability to create unprecedented levels of fear, more so than the traditional methods of scripted triggers. Interaction in our VE plays a key role in the manipulation of agency, providing users with the ability to explore the VE, grab objects and interact with the monster, while letting the environment revoke all these privileges in real time. This means that the environment is designed to give the user control just as it is allowed to take control. This to-and-fro of agency has proved to create higher levels of fear [14]. The VE does this using a variety of different methods and through numerous additions, built on the old environment which only included the scripted AV triggers.

5.1.1 Virtual Hands. The first addition is the fundamental tool that is integral to interaction in VR. That is, virtual hands. These hands are rendered and tracked in real time, and are mapped to the OpenVR inputs in Unity. The user can use the utilize the hands in three core aspects: grabbing objects, controlling the boat, and interacting with the menu, all throughout the experience.

The hand models that are implemented are intermediate-poly (consisting of a moderate number of polygons) and are skinned with a blank white mesh and custom shader. This shader provides the ability to change the color of the hands, enabling inclusivity options for the user. The hands are tracked by use of the vive base stations, and were carefully positioned relative to the main camera (which corresponds to the headset), so virtual hand movement matched real hand movement as closely as possible. The virtual hands and headset camera are sibling GameObjects, so the hands do not inherit the movement and rotation of the headset camera, thus allowing independent hand movement from the users view. Each hand is its own GameObject, each with their own mesh renderer components, but with the same shader – so if the user selects a new hand color, both hands will change at the same time.

The core function of the hands is to interact with GameObjects in the virtual world. This is done by use of the HandGrabbing script which is attached to both the LeftHand and RightHand GameObjects. The HandGrabbing script works by looking up GameObject colliders in the immediate vicinity of the hands by comparing the Vector3 distance (relative x, y, z coordinates) and attaches to a collider and its GameObject with the tag "Grab." The object only attaches if the user also holds down the trigger of the vive controller. This means that we simply tag any GameObject we want the user to be able to pick up or grab in the VE by assigning the tag in the Unity hierarchy throughout development. Once the object has been picked up, we determine if the object has a RigidBody (RB) component (responsible for Unity physics), and if it does not, we instantiate and attach a new RB. Either way, the RB physics are frozen, and the grabbed object is childed to the associated hand. The object then inherits all position and rotation floats of the hand, allowing realistic manipulation of the object. When the user lets go of the trigger, the virtual objects

RB component unfreezes, thus restoring physics to the object. If the user throws the object, the relative velocity of the hands are captured at the moment of object release and assigned to the object, propelling the object in the direction of the throw and letting gravity affect the objects fall.

The only other script that is attached to the hand GameObjects are hand animation scripts. When the user holds the trigger down, an animation is triggered and the hands close as reals hands would when grabbing an object. Each animation is instantiated separately, so if for example the left controller trigger is pushed, only the left virtual hand will play the grab animation.

5.1.2 Non-linear Exploration. The previous environment only contained a linear path throughout the experience. One where the user was simply guided on boat through the canal, with no control or freedom to switch to different canal segments. The new VE was completely reworked to enable non-linear exploration, resulting in an entirely new path the boat follows every time the user wishes to experience the environment, adding a sense of unfamiliarity and fear of the unknown. As a consequence, major changes need to be made to both the layout of the canal map and the controls of the boat.

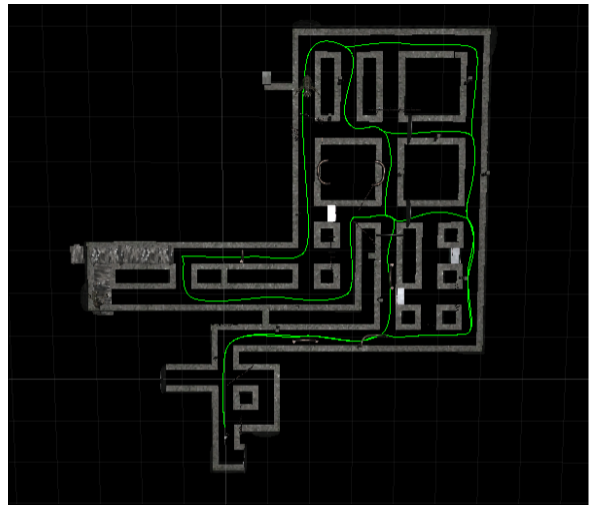


Figure 5: New map with Bezier Curves

5.1.2.1 The Sewers Map. The map of the various canal segments changed drastically in order to accommodate diverging paths and non-linear gameplay. We started by designing a new layout by creating a paper prototype of the birds-eye view of the new map. This map depicted new segments that needed to be created and with subsequent paths that diverge through each relevant canal. It was decided that the new map needed a total of 4 paths, and each path could intersect and reconnect at different points of the map (see figure 5). To achieve functionality in a practical sense, we decided to use multiple Bezier Curves; as this provided an intuitive and clear method of making the boat follow a set path. The boat GameObject was attached to the curve itself, and a Bezier curve PathCreator plugin allowed us to retrieve the current position of the boat at any point of the curve, returning a float value of the boats precise position at any time. It also allowed the boat to align to the quaternion rotation of the curve while on the path (which accounts for boat rotation). All four paths are overlayed on each other, and intersect when diverging. The idea was that if the user decides to switch paths at a canal segment intersection, since they are overlayed, the boat will switch paths without the user noticing any position snapping – which could lead to loss of immersion. For instance, at the first intersection, the user can switch from path one to path two, but if the paths were not overlayed, the user would notice the boat teleport to other path if their float values were even slightly different. The problem still remained that there were roughly twelve diverging paths, so retrieving input at each precise divergence is difficult. We implemented a decision tree that helped streamline this process (see figure 6).

The decision tree has two core conditions that allows the boat to switch paths. The first is the location of the path switch. Since we can get the position float of the boat on the Bezier curve at all times, we got an approximate interval of floats at each of the twelve diverging branches. When the user is within that interval, the path switch can occur, and the float value is assigned to the new path of the boat. The next condition is the input of the user. We wrote a RockBoat() method and Boat script that retrieves the touchpad input form the vive controller that the user can press. When the user presses one of the directions of the touchpad, the direction input is retrieved and the boat switches paths. The boat also subtly rocks forward, left or right at a slight angle to provide visual feedback while in the experience. The combination of the float position and the input from the user determines which path to switch to, and proved to be very successful in practice.

5.1.2.2 The Boat. In an effort to change the constant speed of the guided boat in the previous VE, the new boat was developed with interaction in mind. Apart from switching paths with the touchpad, a paddle was implemented that allows the user to physically row the boat to move it on the Bezier paths. Since the boat follows the path, the user cannot use the paddle to turn the boat, only propel it forward. This creates less of a distraction for the user, who would need to focus on other aspects of the experience. The paddles uses a script to check collisions with the water, which increments the speed of the boat. The speed is then decremented slowly until it comes to a halt. To delay the user spamming the water for too much speed, a coroutine was used to create a delay on collisions with the water. If the user drops the paddle at any time, the paddle will respawn after approximately five seconds, adding some tension if the user loses their paddle. There is also a placeholder paddle, a semi-transparent white mask of the paddle that is also visible at all times. If the user drops the paddle over this mask, it will anchor to the side of boat, allowing the user to pick it up again.

5.1.3 Torches. Torches are the addition to the environment that has the capacity to induce high levels of fear. The torches are GameObjects that are parented to a torch holder, which is parented to a brick column that resembles the aesthetic of the rest of the sewers (see figure 7). These were strategically placed throughout each canal. The purpose of the torches is the manipulate the agency of the user, while providing a tone of warmth to the experience and juxtaposition to the surrounding darkness. The objective is for the user to *need* the torch in order to feel safe – and we developed certain methods that ensure this only *sometimes* happens. The torch is tagged with “Grab”, meaning it is the only object other than the paddle that can be picked up by the virtual hands. The primary purpose of the torch, other than a light source, is to scare the

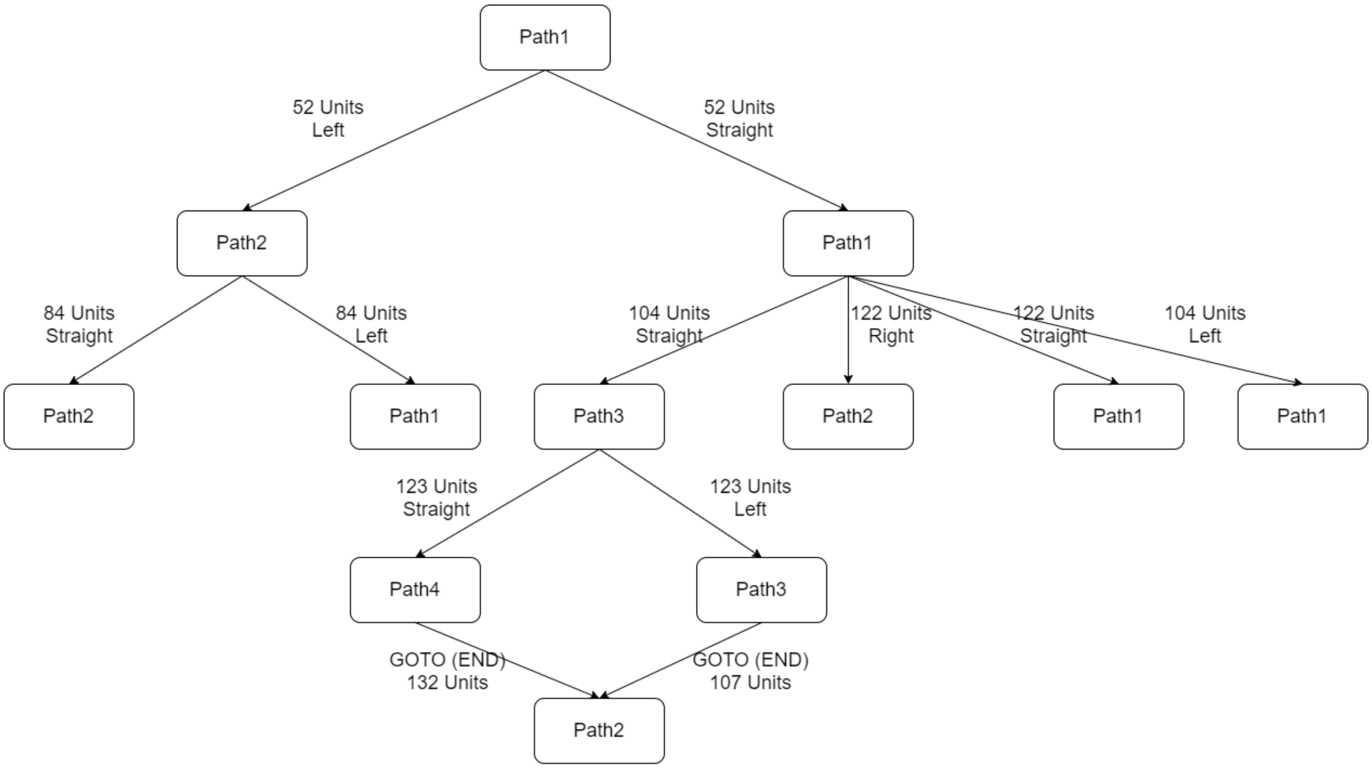


Figure 6: Decision tree for branching paths

Figure 7: Torch and torch holder

monster if the monster finds the user at any time during their experience. The monster AI checks if the player is holding the torch, and, if so, retreats and produces an audible scream. If the player does not have a torch in hand, the monster will fill the viewport of the user's camera, and roar loudly. The torch does not respawn if dropped, so torches are valuable assets to the user – as without a torch, it is completely dark. Finally, the torch AI is among the most advanced in the environment, which is elaborated in the AI section (see 5.2.1).

5.1.4 The Radar Device. The radar device is a small device that resembles a watch, attached to the right virtual hand of the user. The primary function of the device is to develop tension, anticipation and fear of the unknown. There are two blips on the radar. One is the user, which is a green cursor permanently resides in the centre of the radar. The other is a red, pulsating, circular blip that changes position in real time in relation to the boat (see figure 8). The radar rotates inversely from the rotation of the parent GameObject, the right virtual hand, which ensures that the cursor in the middle is always facing forward when the user looks at their wrist to check the monster's position. The radar device is only available in interaction level 3.

5.1.5 Interaction Levels. To contrast each level of interaction, and create a foundation for future work where we could properly analyse this contrast, three levels of interaction are available to the user. The purpose is to define and declare a hypothesis that level three (all interaction and AI) should ultimately evoke more fear than level one (no interaction or AI). It is worth noting that all levels of interaction includes all the previous work done in the previous iteration of the environment; that is, all scripted visual and audio

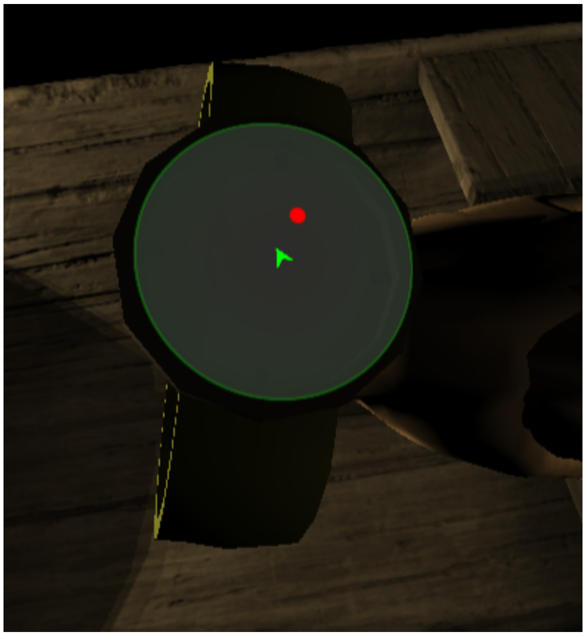


Figure 8: Radar device

Figure 9: Monster lunging from darkness

triggers. The interaction levels can be chosen at the main menu prior to entering the virtual environment. The interaction level choice by the user is recorded using PlayerPrefs – Unitys form of persistent data across multiple scenes – that is referenced in the AI director, which determines which GameObjects and scripts should subsequently be disabled in the specific environment the user wishes to experience. For comparison of these levels, and exactly what is enabled and disabled, see supplementary table S1.

5.1.6 Menu and Tutorial. The previous environment lacked any cohesion in terms of enabling disabling triggers. The user had to remove the headset and use the mouse to click toggles on-screen, during the experience, which leads to loss of immersion. We designed a whole new menu that configures the environment prior to each interaction level, for professionality and accessibility. The menu uses a red and black colour theme that corresponds to the overall theme of fear; and uses the Helvetica typeface for clarity and visual appeal. The virtual hands and the vive controller triggers are used for intuitive interaction with the menu using a pointer that is emitted via a raycast from the right hand GameObject. On this menu, the user can view controls, set skin colour, set interaction level, and access the tutorial.

The tutorial is a mini environment that resembles the sewers but with directional lighting on. It contains an introduction using various screens that tells the user how the interaciton, the various mechanics, controls and monster works prior to the actual experience. It is recommended on the main menu that the user *must* complete the tutorial before the experience for safety and usability reasons.

5.2 Artificial Intelligence

5.2.1 Monster AI. The monster AI is responsible for the main source of fear: the spawning and pathfinding of the monster and its ability to confront the user. The monster AI is a script called FollowPlayer which is attached to the “monster 1” GameObject. The high-level overview of the script is as follows. The script is based on a probabilistic model; one that relies on chance to reach the user. The monster spawns at a random distance interval from the boat, on one of the four paths, and with a random running speed. It uses the Bezier curve as the boat does to move along the path. The monster can spawn in front or behind the user, and has a chance to “stalk” the user, which decrements the monsters speed, to create an illusion of being followed. In conjunction with the radar device, this proves to be effective in developing tension. If the monster is close to the user, but the user has a torch, the monster flees with a distinct fleeing noise and then resets. If the user does not have a torch, the monster roars with a choice of two different roaring noises, and the boat freezes (see figure 9). The users view then fades to black and fades back in, now with the monster respawning and gone. All of this, spawning, stalking and changing paths, happens every 10 seconds in a Respawn() coroutine. Resetting the monster involves being in “limbo” for 10 seconds before respawning, giving the user some breathing room to get ready for if the monster returns. This AI has been tailored and checked to scare the user at least twice but not more than four or five times during the experience. This AI is independent of the other triggers that involve monsters, which instantiate their own monster models. This behaviour is available on interaction levels 2 and 3 only.

5.2.2 The AI Director. The AI director is responsible for controlling the dynamic elements of the environment; ones that change using numerous coroutines, controlling the users fear. The AI director contains a number of scripts, which are components of the “Scene AI Controller” GameObject. More basic scripts control the fundamental aspects of the experience: how the experience ends, which scripts and GameObjects are enabled or disabled depending on the interaction level, how the boat attaches to the Bezier curves, which torches spawn at certain spawn points and fading UI controllers for transitions if the monster scares the player and when the experience fades to a close. However, the torch AI (called LightController), also controlled under the same GameObject, is the most complex, and is responsible for the strongest manipulation of user agency.

The torch AI script contains three coroutines that all run simultaneously, and from the start of the experience. Each coroutine contains a List<> constructor that contains a dynamic array of all torch GameObjects. The first coroutine simply enables and disables a random torch at random intervals of around 10 seconds. The second coroutine looks up the Vector3 distance of each torch in relation to the user, and if under a certain distance, begins a sequence. The sequence has a 50/50 chance of executing, and if this condition is met, flickers the torch on and off for a few seconds until the torch is disabled. When this occurs, the "Grab" tag is disabled, and the user is forced to observe the torch disappear before their eyes. When the torch begins the flicker, a wind-tunnel sound effect fills the canal, followed by the sound of something blowing out the torch. This effect is extremely effective; and is one of the more powerful fear-inducing effects. The last coroutine does another Vector3 lookup, but in relation to the monster. If the monster is close to a torch, the torch goes out. So if the monster is running towards the player, there is a visible stream of torches suddenly going out, creating a very frightening sight. Finally, if the user picks up a torch, the torch will inevitably go out after around 40 seconds with the same "blowing out" sound effects, forcing the user to look for more torches. This functionality is attached to the HandGrabbing script on the virtual hands themselves. All of this torch behaviour is only available on interaction level 3.

The AI director also spawns barriers depending on which path the user is on. This is only to prevent the user from changing paths and going backwards down a canal, avoiding the end of the experience. Barriers spawn if the user collides with certain triggers that change location. These triggers are located on the paths themselves.

6 EVALUATION

Prior to the onset of SARS-CoV-2 in South Africa, user-testing was the focus of our experimentation. This kind of study is still the preferred method of testing in a project such as this; as the autonomic nervous system of the individual, in a large sample size would have been extensively tested, providing conclusive results for the elicitation of fear using interaction. Due to the uncertainty of the changing COVID climate, we opted for three expert heuristic evaluations of our environment. First outlined by Molich and Nielson [31], expert heuristic evaluations remain effective for testing purposes even with small group (three to five people), which adequately met our testing requirements. Additionally, we ran performance tests to determine if the VE performs optimally in terms of framerate, ensuring the user is comfortable and does not require a top-tier graphics card to run the simulation. We propose that this method of evaluation be a pilot study for this area of research; and act as a starting point for evidence that interaction is successful in eliciting fear over traditional scripted cues in virtual environments.

The expert evaluation was conducted with two phases in mind. Each phase involved completion of a form that relates emotions and usability of the environment to the user. These forms were given to the evaluators to complete over a few days. Completed forms were sent back to us at the end of the testing period, and a latest build executable of the environment was provided to them for these tests. The evaluators were to use their own OpenVR equipment, and were instructed to employ a room-scale setup due to the inclusion of full tracking of the headset and the virtual hands, and the need for greater mobility – as the environment can on occasion require moderate spatial movement.

The first phase was that of an open form questionnaire, asking the evaluators open questions regarding their current emotions, state-of-mind, VR setup, the interaction levels and what they preferred in each environment. It was instructed that these questions should be answered in as much detail as possible for production of accurate results. This form was to be completed before and after the environment, to develop a contrasting effect for better discussion of the results.

The second phase involved an expert heuristic evaluation form intended to identify the impact of usability problems in the environment. This form was based on the VR heuristic evaluation protocol, outlined by Sutcliffe and Gault [32], for adaption of traditional heuristic evaluation in VR. The structure of this form is based on three categories. The first is design class problems, where the evaluator is asked to identify elements in the environment that resembled a usability issue under a design class. An example of this would be a problem: the monster stops respawning, and the design class: Artificial Intelligence. Once the problem has been identified, the second category, heuristics, are assigned to the problem from the list of given heuristics. In this case, consistency or a custom heuristic could be used, as the monster does not reappear. Finally, a severity rating, ranging from 0-4, will be given to this problem, where 0 represents a very minor problem, and 4 is a very severe problem that needs to be fixed immediately. The monster not spawning would be a 4, as the entire fear experience is dependent on this use case.

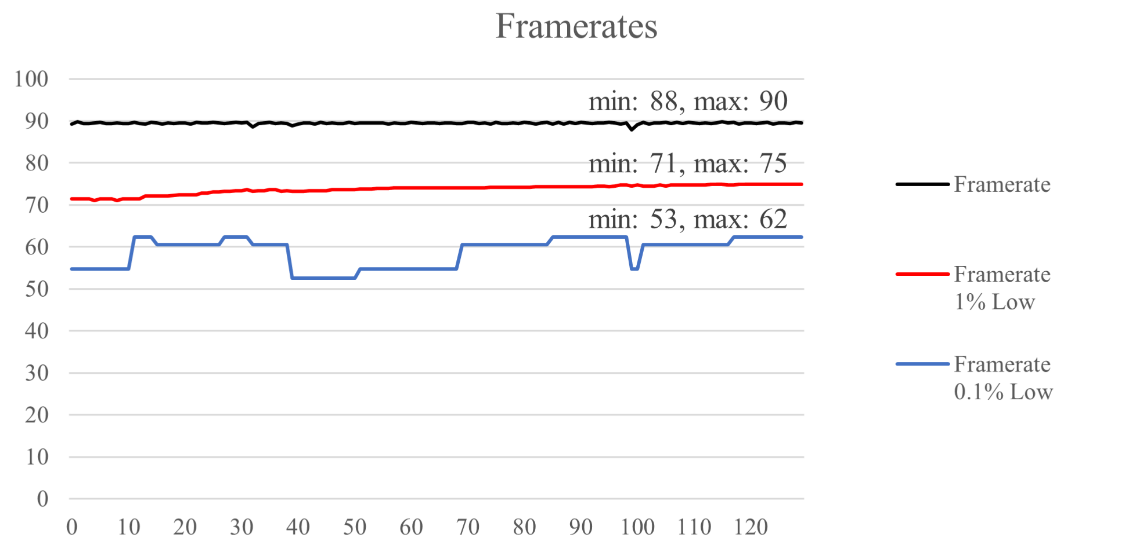
To aid the evaluator in remembering what they experienced, text logs of interaction and behaviour of the environment as a result of the user and the AI director are generated during the experience. These logs record the exact time of each event. For example, if the monster scared the user, this event would be available to review in the log after the experience.

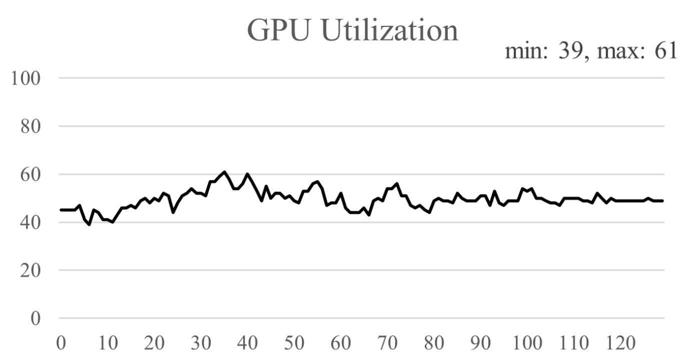
7 RESULTS AND DISCUSSIONS

Due to the nature of the evaluation, the project has shifted to a pilot study; one where there is a focus on a heuristic usability analysis of the VE. This is to prepare for future iterations of the project and environment, when proper user-testing experiments can occur. We are laying the foundations for an environment that has the potential to elicit higher levels of fear with interaction. However, the open forms give us insight and valuable feedback of the success of interaction and AI.

7.1 Open Forms

The open forms yielded interesting yet expected results. The objective was to determine if interaction level 2 or 3 elicits higher levels of fear as opposed to level 1. All three expert evaluators confirm that interaction did just this; there was notable higher levels of fear evocation in level 3 according to their personal opinions. All three evaluators wrote that the intensity of their emotions had a definite increase while experiencing the environment. More unexpectedly, all three evaluators described different elements of the environment to be the source of this higher level of fear. And, confirming results that manipulation of agency [14] can evoke a stronger emotional response, these elements were tied to agency.





Figures 10 and 11: GPU Usage (% versus time (s)) / Framerate (fps versus time (s))

The torches; the only source of defense against the monster, proved to be effective. Allowing the freedom of grabbing a torch, but then making the torch flicker and disappear was a great source of tension, while the anticipation of not being able to grab the torch while the monster lurks equally increased tension. Even more effective were the unexpected jump scares; and the fear of not knowing where the monster was. Stalking was successful in this sense; confirming that the monster AI was sporadic enough that it developed anticipation and tension without too much redundancy.

An evaluator also found that the radar and paddle were their favourite elements of their environment. This was because the manipulation of user control was present. Allowing the user to move through the environment at their own speed, while remembering to look at the radar to locate the monster in the darkness can create a feeling of being overwhelmed by many tasks. This in itself can cause more anxiety, fear and stress. The same evaluator, however, had a preference for interaction level 2; as the concurrent tasks made it *too* overwhelming, and thus resulted in a loss of immersion. This can confirm that the balance of immersion in the environment is crucial to emotional response.

We can be certain that, based on the opinions of the evaluators, one can experience a stronger sense of fear in a VE with the presence of interaction. See supplementary section open forms for the aforementioned answers from these evaluators.

7.2 Performance

An important metric in VR to determine if an environment performs optimally, is framerate. Consistent framerate is integral to the overall experience for the user, as frame judder – a result of inconsistent jumps in framerates – can cause motion sickness and a loss of immersion. We ran the environment for two minutes on a low to medium tier VR capable graphics card, the GTX 1070, and determined average framerates, 0.1% and 1% lows and the temperature of the GPU. We used RivaTuner and MSI Afterburner to log each measurement throughout a full playthrough (approximately ten minutes) of the environment at Ultra settings and 1080p resolution. These settings enable full anti-aliasing, texture resolution, ambient occlusion and other quality enhancements. See below figures for results (figures 10 and 11).

The results of our performance tests are extremely favourable. Our environment does not even utilize half the maximum performance - an average of 49% usage - of the GTX 1070, and remains at a consistent 90 FPS with an average framerate of 90 FPS (a and b). This means that the recommended 90 FPS target to synchronize with the 90Hz display of the HTC Vive is reached, resulting in a smooth experience for the user, with almost no judder. This will significantly reduce simulator sickness as a result of frame consistency and almost no frame drops.

The 1% and 0.1% lows (the absolute worst framerates at 1% and 0.1% of the entire benchmark) average 73 FPS and 58 FPS respectively, which is still near 60 FPS, which in itself results in smooth motion. The frame window for these metrics are also small at 1.2 seconds and 120ms total runtime respectively.

It can be assumed that if a user has a system that is capable of running an HTC Vive, they can run the VE without major framerate issues, resulting in a smooth experience without motion sickness from frame drops. It is worth noting that the GTX 1070 is a medium tier graphics card and considered low tier in the field of VR. This benchmark was also conducted on an i5 4690, a relatively low tier CPU as well, especially for usage with VR. This can confirm that the VE is accessible to most VR users.

7.3 Heuristic Evaluation

The heuristic evaluation serves as a key indicator of usability problems that should be considered for future studies with this environment. To accurately determine severe issues, we examine usability problems with a severity of 3 (of a max 4) or above in the table (see table 1 below).

Heuristic	Problem	Frequency
Realism	Virtual hands do not accurately follow actual movement of hands	3
Interaction	Paddling motion unnatural	2
Interaction	Speed of boat hard to control	1
Realism	Unrealistic monster behaviour at times	2
Realism	Fear elements should be left out of tutorial and menu	1
Realism	Could not reach the torch when I should be able to	1
Simulator sickness	Jerking of the boat induced sickness	2
Interaction, agency	Boat rocking and path switching not effective enough	1
Consistency	Menu button ray casting	1

Table 1: Usability problems (≥ 3 severity)

Following this data and feedback, we immediately developed a new build of our project, fixing some of the major usability issues listed above, such as the orientation of the hands, menu usability, the tutorial and being able to reach the torches.

Simulator sickness, however, remains a severe issue affecting the majority of the evaluators. This could not be addressed during project development due to time constraints, the magnitude of the issue, and because the issue was not apparent before preliminary testing at the final stage of the project. The boat was fundamentally developed with the intention of not causing SS, and, since we did not experience any such sickness ourselves, we were unaware. This issue in itself could be investigated in future studies that aim to reduce motion sickness by use of a novel boat motion technique, rather than simple boat mechanics.

We can now safely justify that, with the exception of SS, the changes that need to be made are minor changes rather than massive feature releases or bug fixes, and the project is adequately prepared for future study.

8 CONCLUSIONS

Our primary focus of our new VE was to utilize interaction and AI as powerful methods of eliciting fear. Our expert evaluators all agreed that this is possible with our environment, but lack of extensive evidence dictates that this project needs to be iterated further. Interactive elements, such as dynamics of the torches and the radar device in the VE, which were associated with user agency, evoked a higher sense of immersion and fear, which is promising. The monster and its AI proved to be successful in creating believable jump scares and stalking the user created an interesting dynamic in developing tension in the user. Further development of interaction and AI in the environment should hone the map layout of the sewers and improve the behaviour of the monster. Interaction with the monster itself should also be a priority.

We were ultimately successful in identifying usability problems that needed to be fixed, and we created a VE that performs optimally, even on low to medium tier VR computer hardware. However, improvements can be made to ensure optimal user comfort, to avoid simulator sickness. This was due not to performance, but rather the motion of the boat – an integral part of the experience. Investigation into novel techniques of boat motion, while reducing sickness should be a priority in future development of the VE.

These aspects label our VE as a reliable medium for future user-testing, which should involve autonomic nervous system tests, such as respiratory, heartrate and skin capacitance metric recordings. User-tests should involve a sample size of over twenty subjects and should be conducted on a newer iteration of the environment that aims to fix smaller issues. Results of this test would undoubtedly provide a definite answer as to whether interaction and AI is the more effective method of eliciting fear, over that of traditional scripted triggers.

ACKNOWLEDGMENTS

We would like to thank multiple people who were involved in development of the fear virtual environment for VREmote. We thank Brent van Der Walt for some of his input on the design of the environment and logistics of the project. We thank Gosia Lipinska and Siphumlele Sigwebela from the psychology department for their continuous support and ideas for each iteration of the environment during the development process, from a non-computer science standpoint. Finally, we thank Assoc. Prof. James Gain, our supervisor, for guiding us through each and every complication of this project, especially during the peak of the global COVID-19 pandemic.

REFERENCES

[1] Craske, Michelle G., Michael Treanor, Christopher C. Conway, Tomislav Zbozinek, and Bram Vervliet. 2014. Maximizing exposure therapy: An inhibitory learning approach. Behaviour research and therapy 58: 10-23.
[2] Steuer, Jonathan. 1992. Defining virtual reality: Dimensions determining telepresence. Journal of communication 42, no. 4: 73-93.
[3] Rizzo, Albert, Jarrell Pair, Peter J. McNerney, Ernie Eastlund, Brian Manson, Jon Gratch, Randy Hill, and Bill Swartout. 2005 Development of a VR therapy application for Iraq war military personnel with PTSD. Studies in health technology and informatics 111: 407-413.
[4] Wu, Di, Dongdong Weng, and Song Xue. 2016. Virtual Reality System as an affective medium to induce specific emotion: A validation study. Electronic Imaging 2016, no. 4: 1-6.

[5] Rothbaum, Barbara Olasov, Larry Hodges, Samantha Smith, Jeong Hwan Lee, and Larry Price. 2000. A controlled study of virtual reality exposure therapy for the fear of flying. Journal of consulting and Clinical Psychology 68, no. 6: 1020.
[6] Powers, Mark B., and Paul MG Emmelkamp. 2008. Virtual reality exposure therapy for anxiety disorders: A meta-analysis. Journal of anxiety disorders 22, no. 3: 561-569.
[7] Henrik M. Peperkorn, Julia Diemer, Andreas Mühlberger. 2015. Temporal dynamics in the relation between presence and fear in virtual reality, Computers in Human Behavior, Vol. 48. 542-547 DOI: https://doi.org/10.1016/j.chb.2015.02.028.
[8] Peperkorn, H.M. and Mühlberger, A. 2013. The impact of different perceptual cues on fear and presence in virtual reality. Annual Review of Cybertherapy and Telemedicine. 75.
[9] J. Hvass, O. Larsen, K. Vendelbo, N. Nilsson, R. Nordahl and S. Serafin. 2017. Visual realism and presence in a virtual reality game. In 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), Copenhagen, 1-4.
[10] Slater, Mel, and Sylvia Wilbur. 1997. A framework for immersive virtual environments (FIVE): Speculations on the role of presence in virtual environments. Presence: Teleoperators & Virtual Environments 6, no. 6: 603-616.
[11] Mütterlein, Joschka. 2018. The three pillars of virtual reality? Investigating the roles of immersion, presence, and interactivity. In Proceedings of the 51st Hawaii international conference on system sciences.
[12] Tooby J., Cosmides L.. Barkow J. H., Cosmides L., Tooby J. 1992. Psychological Foundations of Culture, The Adapted Mind. New York, Oxford University Press (pg. 19-136)
[13] Sidebottom, A. & Tilley, N. 2008. Evolutionary Psychology and Fear of Crime. Policing. 2 (2), 167–174.
[14] Lin, J. -H. T. 2017. Fear in virtual reality (VR): Fear elements, coping reactions, immediate and next-day fright responses toward a survival horror zombie virtual reality game. Computers in Human Behavior. 72350–361.
[15] Orr, S.P. 1990. Psychophysiological studies of PTSD. in: E. Giller (Ed.) Biological Assessment and Treatment of PTSD. American Psychiatric Press, Washington, DC; 137–157.
[16] Abrash, M., 2014. What VR could, should, and almost certainly will be within two years. Steam Dev Days, Seattle, 4.
[17] Gonzalez-Franco Mar, Lanier Jaron. 2017. Model of Illusions and Virtual Reality. Frontiers in Psychology. 1125. DOI: https://doi.org/10.3389/fpsyg.2017.01125
[18] M. Slater. 2009. Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments. Philosophical Transactions of the Royal Society B: Biological Sciences, 364, 3549-3557
[19] Hidaka, H. Qin and J. Kobayashi. 2017. Preliminary test of affective virtual reality scenes with head mount display for emotion elicitation experiment, 17th International Conference on Control, Automation and Systems (ICCAS), Jeju, 325-329.
[20] Mori, M., 1970. The uncanny valley. Energy, 7(4), 33-35.
[21] Wu, D., Weng, D. and Xue, S. 2016. Virtual Reality System as an affective medium to induce specific emotion: A validation study. Electronic Imaging, 1-6.
[22] Chen, Hao and Dey, Arindam and Billinghurst, Mark and Lindeman, Robert W. 2017. Exploring Pupil Dilation in Emotional Virtual Reality Environments. In Proceedings of the ICAT-EGVE - International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments, November 22 – 24, 2017, Adelaide, Australia 169-176.
[23] Bergmann, Till, Matthias Balzer, Torsten Hopp, Thomas van de Kamp, Andreas Kopmann, Nicholas Tan Jerome, and Michael Zapf. 2017. Inspiration from VR Gaming Technology: Deep Immersion and Realistic Interaction for Scientific Visualization. In VISIGRAPP (3: IVAPP), pp. 330-334.
[24] Ohyama, Seizo, Suetaka Nishiike, Hiroshi Watanabe, Katsunori Matsuoka, Hironori Akizuki, Noriaki Takeda, and Tamotsu Harada. 2007. Autonomic responses during motion sickness induced by virtual reality. Auris Nasus Larynx 34, no. 3 (2007): 303-306.
[25] HTC. 2020. What is the recommended space for the play area? https://www.vive.com/eu/support/vive
[26] Norman, Donald A. 1988. The psychology of everyday things. Basic books.
[27] Toprac, Paul, and Ahmed Abdel-Meguid. 2011. Causing fear, suspense, and anxiety using sound design in computer games. In Game sound technology and player interaction: Concepts and developments, pp. 176-191. IGI Global.
[28] Matt and Ross Duffer, Stranger Things. 2016. USA: Netflix, Television.
[29] Protofactor Inc. 2020. LACODON. https://assetstore.unity.com/packages/3d/characters/creatures/lacodon-19167
[30] Hamandishe Mathivha. 2019. Fear in Virtual Reality. University of Cape Town. Department of Computer Science.
[31] Nielsen, Jakob, and Rolf Molich. 1990. Heuristic evaluation of user interfaces.In Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 249-256.
[32] Sutcliffe, Alistair, and Brian Gault. 2004. Heuristic evaluation of virtual reality applications. Interacting with computers 16, no. 4: 831-849.

SUPPLEMENTARY INFORMATION

The section contains additional images and figures to visualize our development decisions and aid understanding of our project. These are appended here due to space limitations.

Figure S1: Legacy XR Input System

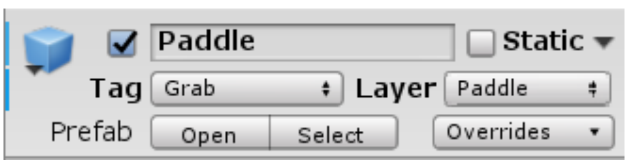


Figure S3: Grab tag on paddle GameObject

Figure S5: Old map layout

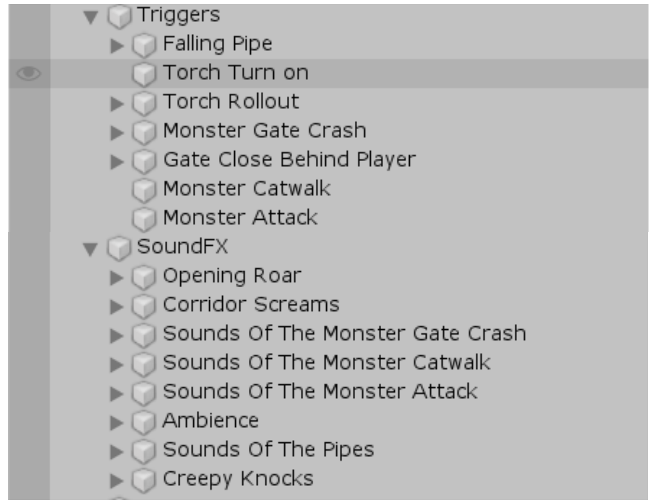


Figure S2: List of AV triggers

Figure S4: Semi-transparent placeholder paddle

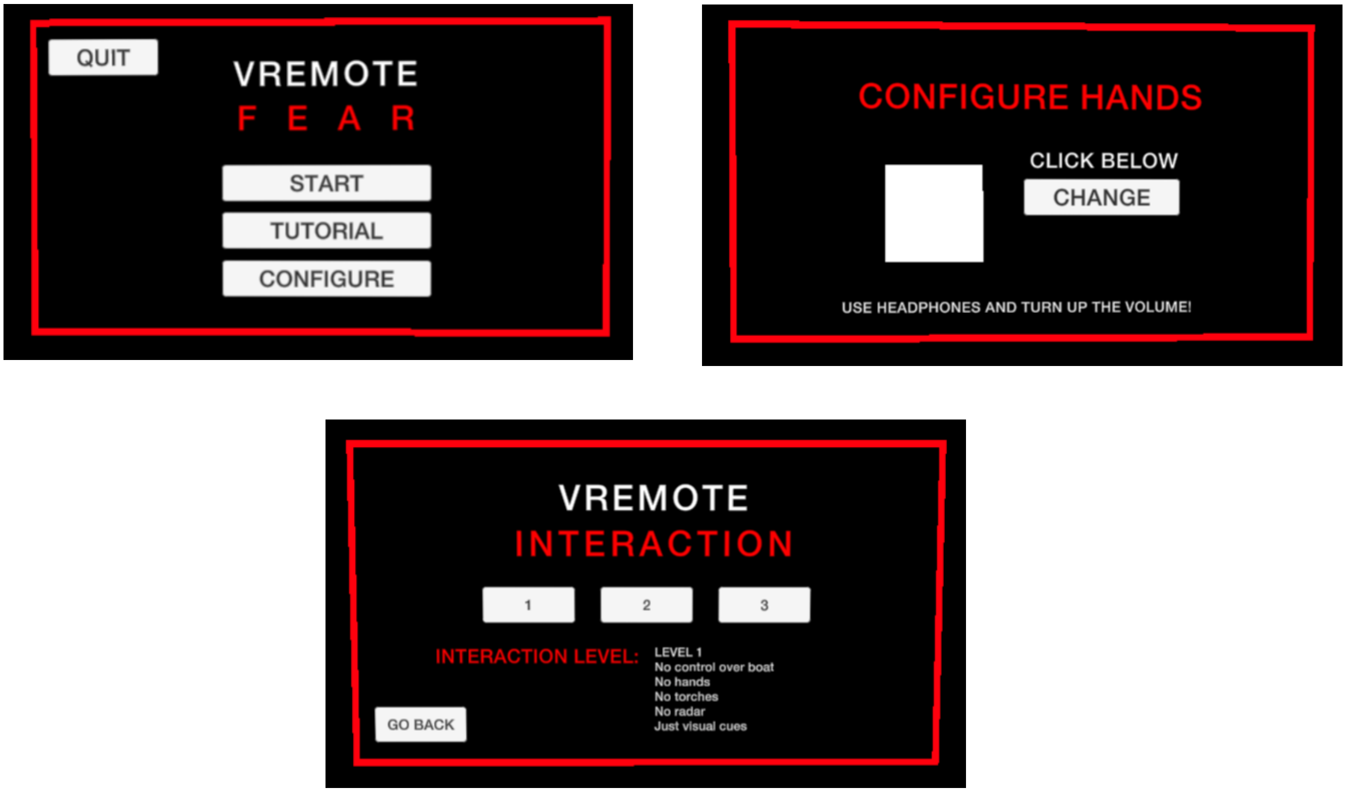


Figure S6: Menu screens

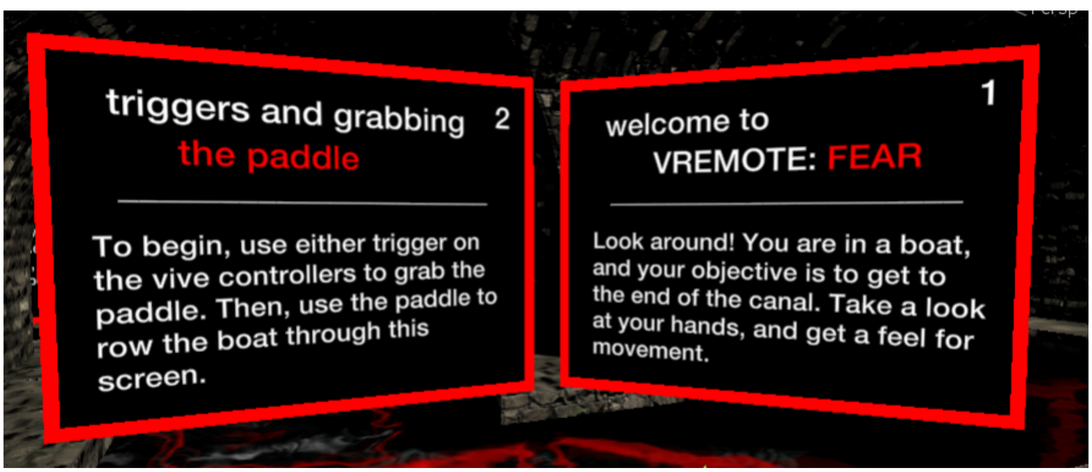


Figure S7: Example of tutorial screens

INTERACTION LEVEL 1	INTERACTION LEVEL 2	INTERACTION LEVEL 3
No control over the speed of the boat; linear throughout the experience. Boat follows a predetermined trail throughout canals.	Speed of the boat can be changed. Boat follows a predetermined trail.	Full control over boat, player can decide to turn down different canals.
No virtual hands	Virtual hands are present	Virtual hands are present
No torches	Torches can be used to scare the monster, causing the monster to retreat.	Torches can be used to scare the monster, causing the monster to retreat.
No radar device	No radar device	Radar is present, with full integration with the AI director.
Monster AI is limited, simple visual cues intended to scare the player at certain points of the simulation are triggered. (Similar to original VE). AI director is non-existent.	Simple following the path to the player is implemented. Torch can scare monster, causing monster to pathfind to new location for respawning. Torch AI not enabled.	All AI present. Advanced pathfinding and AI director decision making creates the dynamic and constantly changing environment. Advanced torch AI.

Table S1: Comparison of interaction levels

OPEN FORM QUESTIONS AND ANSWERS

These questions and answers we recorded after the expert evaluators experienced the environment.

What emotion(s) are you current experiencing?

(1) Tension, remains of fear and discomfort from nausea
(2) Enthusiastic but slight increase in tension. Tight chested.
(3) Anxiety, relief

How strong is/are your emotion(s)?

(1) Medium
(2) Medium
(3) Slightly increased from before

How comfortable do you feel?

(1) Relatively comfortable as nausea continues to fade
(2) Comfortable because I know it is a game and can distance myself.
(3) Slightly uncomfortable, after effects of Simulation sickness

Which level of interaction evoked the most emotion or was the most interesting and why?

(1) Only tested on level 3. Sound effects were interesting and effective, added a great deal to the atmosphere. Really liked the fire blowing out sound. (When you can't see your hearing is heightened)
(2) Not done.
(3) The 2nd stage, I was able to focus on the paddling and picking up torches which led the monster interactions to be the scariest. Focusing on my agency while not being overwhelmed by it allowed the monster to creep up on me successfully. Stage 3 felt like an improvement of this, but I felt too overwhelmed at times which took away from the experience compared to the 2nd stage.

What element of the environment evoked the most emotion?

(1) The monster jump scare.
(2) Gates closing behind. Seeing the creature in the distance.
(3) Realizing the monster was behind me stalking me, and turning around to see it

What did you like most about the environment?

(1) The general feel of the environment. All the visuals matched in quality and theme and this contributed to overall immersion.
(2) Stone architecture made it feel like being in a tunnel.
(3) Being able to move the boat by paddling, also the radar feature was great both in making me more scared and giving me a sense of agency.

Did you experience performance issues? If so, what issues?

(1) Framerates were fine.
(2) No.
(3) Small green flicks in the darkness, and once a grey screen due to going out of the sensor's range, otherwise no issues