Intensity and Compositional Prompts in Videogame Soundtracks

Dale Keaveny

A thesis submitted in fulfilment of the requirements for the degree of Master of Music (Composition)

Sydney Conservatorium of Music

December 2022

I declare that the research presented is my own original work and has not been submitted to any other institution for the award of a degree

Signed……

Date……..

Abstract

Videogames are structured by a series of events that articulate various kinds of intensity arcs. These arcs can play a key role in scaffolding music composition and sound design for games, informing the ways sound designers and composers create content in response to pre-planned gameplay events. This thesis proposes two novel analysis techniques for capturing intensity shifts during gameplay – a Pixel Change Ratio (PCR) analysis that measures the rate of visual changes during gameplay, and an Actions-Per-Minute (APM) analysis that measures the rate of user generated input during gameplay. The Gameplay Intensity Framework developed in Chapter Four combines these new analyses with common videogame scoring practices. In Chapter Five, I discuss how this framework was used to compose the soundtrack for the challenge-based videogame *Unsteady VR*, submitted as the major creative work for this Master of Music (Composition) degree. This discussion highlights the compositional challenges and opportunities that emerge when composing against a range of prompts tied to intensity arcs.

Acknowledgements

I'd like to thank my supervisory team Dr Ivan Zavada, Dr David Kim-Boyle, and Dr Christopher Coady for their efforts in supporting my thesis writing.

I'd also like to thank developers Kyle McKee and Tyler Campbell for their permission to use material from *Unsteady VR* in this thesis and their encouragement during my work on the game.

Navigating This Thesis

This thesis includes both a written dissertation and a portfolio of compositions.

The portfolio - a soundtrack for the game *Unsteady VR* – is accessible through the zip folder titled "Keaveny Portfolio.zip" as a list of WAV files.

Chapter 5 provides an in-depth description of the portfolio, including time-coded video annotations to two unlisted YouTube videos:

1. A 'Gameplay Showcase' of *Unsteady VR*
2. A 'Composition Showcase' of the soundtrack played in Logic Pro X

Clicking on the annotations will take you to the relevant point in either video. Each description will include links to the relevant composition showcase and gameplay showcase next to their title in the following format:

(Gameplay Showcase | Composition Showcase)

Table of Contents

Introduction

Videogame designers and composers have long been guided in their work by the different kinds of intensities a videogame might seek to conjure. The narrative arc of a videogame – it's explicit story or implicit rising technical challenges – often provides a map of intensities that sound designers and composers respond to in their creative work. This thesis seeks to expand this map by introducing two new methods for capturing how quickly a videogame scene changes and how much input a player has within a particular videogame scene. These additional data sets establish two different kinds of intensity prompts (one visible, the other invisible) against which composers might offer compositional response. The major creative work for this Master of Music (Composition) thesis – ten music tracks for the videogame *Unsteady VR* – utilises these prompts along with more traditional methods of videogame composition to deliver a score that is tightly aligned to the visible and invisible components of this challenge-based game. This creative work demonstrates how intensity measurements derived from videogames serve the compositional process by more closely linking the composed material to videogame structure.

Videogame Structure and Intensity Arcs

Videogames are generally confined to a structural arc dependent on challenge and/or story. Ernest Adams defines "story-based" games as those which utilise narrative as a key motivator in the game experience.[1] These games that utilise story narratives often draw on well-established pacing conventions from the arenas of literature, film, and television. Mike Lopez compares the narrative structure of videogames with those from television and film through the use of intensity graphs.[2] Lopez's graphs show how action events of increasing intensity unfold over time as a story progresses on a television show and then, in parallel, across a video game level. He first graphs a generic plot timeline to show how intensity ebbs and flows across a rising story arc (*Figure 1*), and then graphs a television show (*Figure 2*) and a videogame level (*Figure 3*) to draw comparisons.

[1] Ernest Adams, *Fundamentals of Game Design, 3rd ed,* Pearson Education, 2014, pp. 81-82.
[2] Mike Lopez, "Gameplay Fundamentals Revisited: Harnessed Pacing & Intensity," *Gamasutra*, 12 November, 2008, https://www.gamedeveloper.com/design/gameplay-fundamentals-revisited-harnessed-pacing-intensity.

Figure 1 - A Generic Plot Timeline (Lopez, 2008)

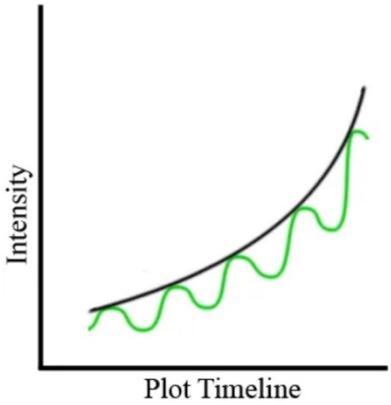


Figure 2 – Pacing Of A Generic TV Episode (Lopez, 2008)

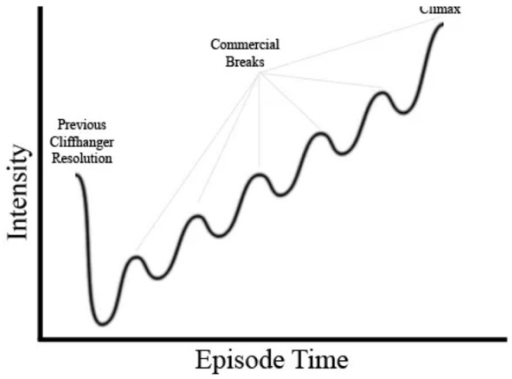
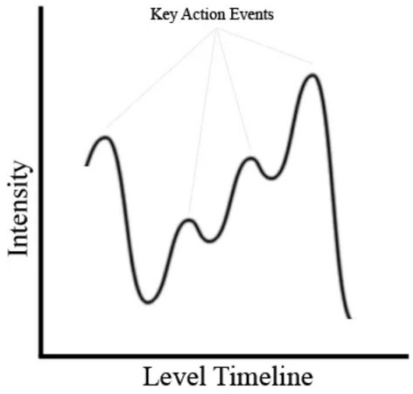


Figure 3 – Pacing Of A Generic Videogame Level (Lopez, 2008)



In story games, narrative drives the gameplay a soundtrack is designed to fit. Story narrative and game soundtrack are thus explicitly linked. Pacing structures that inform the story narrative are used in soundtrack development as early as the pre-production phase, during which composers and designers engage in "spotting" – i.e. planning the basic parameters of compositions before they are recorded for the soundtrack. As Collins explains:

> As with music, spotting the game and creating a list of assets needed is the first step in the sound design of a game. This involves a reading of the script or design document, looking to spot for objects, actions, environments, pace, tension/release, characters/personalities, and so on.[3]

[3] Karen Collins, *Game Sound: an Introduction to the History, Theory, and Practice of Video Game Music and Sound Design*, Cambridge, Massachusetts: MIT Press, 2008, pp. 92.

In arcade or mobile games without an explicit story, the intensity of a game can be paced via gameplay challenge. Adams defines "challenge-based" games as those in which gameplay itself is the focus; based around pure action or strategy with little inclusion of storytelling[4]. Games structured around challenge gradually increasing difficulty over time include the classic arcade games Space Invaders[5] and Tetris.[6] Yet despite the conceptual difference between challenge-based games and story-based games, the narrative of a rise and fall (or simply a constant rise) in intensity can be seen to continue to inform the way sound designers and composers approach a games' sound world. Collins illuminates the long history of this practice: "Space Invaders (Midway, 1978) set an important precedent for continuous music, with a descending four-tone loop of marching alien feet that sped up as the game progressed."[7] In this context, music adheres to intensity structures to maintain a player's engagement in addition to managing their attention. This has become a common practice in sound and music design for games, evident in a wide range of recent rhythm-based music games such as *SingStar,*[8] *Guitar Hero*,[9] and *Beat Saber*,[10] where the player completes timed input challenges to motivate an on-screen performance.

Peter Peerdeman's work on intensity in videogames offers some helpful concepts for understanding the way story games and challenge games overlap in their engagement with intensity.[11] Peerdeman writes: "The dynamic range of a game is the amount of difference between the least intensive moments and the most intensive moments in the game experience."[12] Story games and challenge games might therefore deploy the same dynamic range across a gaming experience even if the events underpinning "the least intensive moments" and "most intensive moments" are radically different. To illuminate the wide applicably of this concept, Peerdeman examines the intensity arcs of a linear shooter game

[4] Adams, *Fundamentals of Game Design*, 83.
[5] Taito. *Space Invaders*. Midway. Atari 2600, 1978.
[6] Nintendo of America. *Tetris*. Nintendo of America. 1989.
[7] Collins, *Game Sound*, 12.
[8] London Studio. *Singstar*. Sony Computer Entertainment Europe. PlayStation 2. 2004.
[9] Harmonix. *Guitar Hero*. Red Octane. PlayStation 2. 2005.
[10] Beat Games. *Beat Saber*. Beat Games. Microsoft Windows. 2019.
[11] Peter Peerdeman, "Sound and Music in Games by Peter Peerdeman" (2010), *VU Amsterdam.*
[12] Peerdeman, *Sound and Music*, 12.

(*Figure 4*) and an open-world game (*Figure 5*) and shows how musical intensity often closely follows the journey through gaming intensity a player encounters in games of different sorts.

Figure 4 – Peerdeman's Intensity Arc For A Linear-Shooter

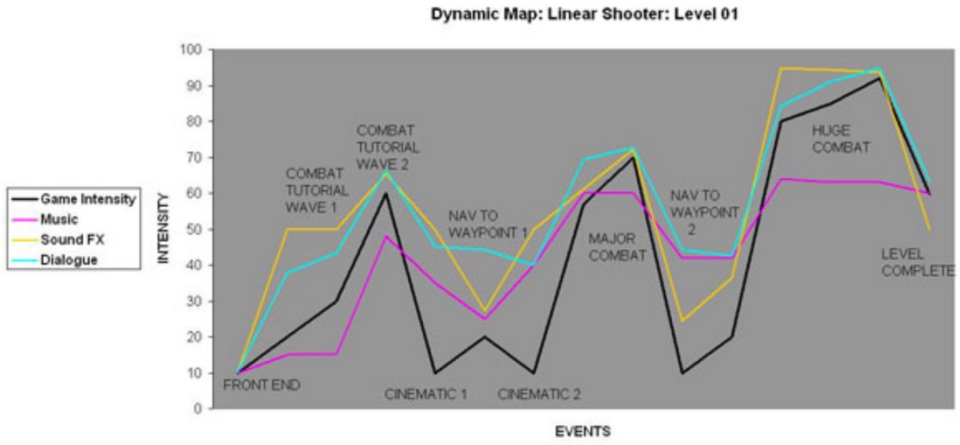
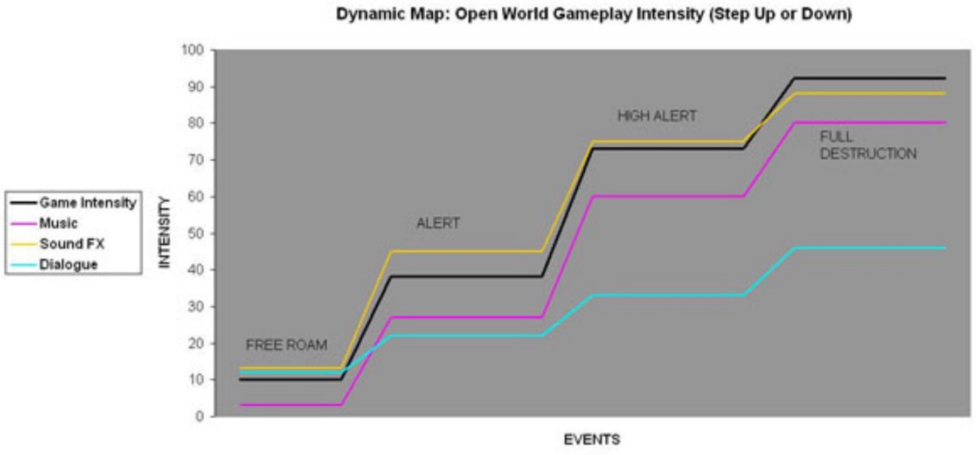


Figure 5 – Peerdeman's Intensity Arc For An Open-World Game



Evident in the accounts of intensity above is that the experience of gameplay looms large in the way music tracks are tailored to fit various kinds of videogames. In this thesis, I synthesise existing compositional prompts in videogame design with new methods of mapping game intensity in order to develop a new framework for videogame composition. This process

unfolds across four chapters. In Chapter 1, I map out the normative procedures videogame composers utilise in making their tracks fit the games they are asked to score. In Chapter 2, I discuss how new developments in the analysis of videomusic – an artform distinct from videogame design – can be used to usefully parse the range of musical events that occur within videogame play. The analytical language developed for videomusic is then brought into a discussion of videogame narratives in Chapter 2 to deepen our understanding of the prompts videogame composers already use. In Chapter 3, I present two novel compositional prompts which directly link musical expressions to intensity shifts parsed from gameplay. In Chapter 4, I combine these prompts with the analytical language from Chapter 2 to form the Gameplay Intensity Framework, detailing a method composers might use to harness the intensity arcs that can be derived from gameplay footage and user input when undertaking creative work. In Chapter 5, I demonstrate how this framework was used to compose and integrate the soundtrack to the game *Unsteady VR*. This practical demonstration of concepts highlights the compositional opportunities that emerge when composing against a range of prompts tied to intensity arcs.

Chapter 1

Musical Mindsets and Compositional Prompts in Video Game Design

In *Game Sound,* Karen Collins casts a wide net around the kinds of musical thinking at play in videogame soundtrack design.[13] For Collins, sound design and composition follow three distinct phases of game design: Pre-Production, Production, and Post-Production.[14] Composition is thus cast as an iterative process in Collins' work – the interweaving of musical sounds with other aspects of game design, refined in stages, is shown to eventually add up to more than the sum of its parts. In this chapter, I unpack Collins' investigation of videogame design as a way of establishing a baseline for how compositional prompts and responses are normally utilised. Despite the publication date of Collins' work, the definitions and concepts explicitly related to the design phases of videogame music hold true when compared to recent research. Gina Zdanowicz and Spencer Bambrick's detailed overview of the audio development process in *The Game Audio Strategy Guide: A Practical Course*[15] describe the same phases of audio production referenced in *Game Sound* and provide updated references to the terminology utilised by Collins.

1.1 Pre-Production

The early stages of music creation during the pre-production phase of videogame development revolve around design documents which explain the design and implementation of a game's audio. During this phase of videogame development, design documents explain the intended structure and creation process for every facet of the videogame. An audio design document is created to explain how audio will be designed and implemented, assisting the audio team and programmers in making choices during the production and post-production phases. While the detail and specificity of these documents varies between games, the general objective of this phase of planning is to determine stylistically appropriate music and sound.

[13] Collins, *Game Sound*, 89.
[14] Collins, *Game Sound*, 89.
[15] Gina Zdanowicz, and Spencer Bambrick. *The Game Audio Strategy Guide: A Practical Course*. Taylor and Francis, 2019.

> The game design document represents the overall vision of the game, detailing the storyline, dialogue, maps, audio, graphics, animation, and programming. The entire team—including the audio team—works from this document during the production process.[16]

Temp tracks, cue sheets, and diagrams depicting emotional tension and release points are common tools used to determine the aesthetic of the music during the pre-production phase. These tools can align the aesthetics of the music, sound effects, and dialogue prior to composition. The pre-production process can also determine a full list of music assets[17] required for the final game, informing scheduling and the organisational steps required for any recording or external production assistance. Furthermore, this phase influences the means through which audio will be integrated and presented by the game. Choices made here regarding audio implementation can have far reaching consequences that extend into production and post-production phases.

1.2 Production

During the production phase of videogame development, audio tasks are mostly related to creating and implementing the assets themselves. Music, dialogue, and sound effects are recorded directly or sourced from third-party asset libraries to fulfill the requirements detailed in the design documents. These assets are then inserted into the game using solutions determined during pre-production. This process is referred to as implementation (or integration) and makes up a significant part of the audio production process: "Integration typically determines how audio will be cued or triggered in a game, as well as what aspects of the audio may be changed by the game state or game parameters."[18] The importance of integration stems from the triggers which control the final presentation of the soundtrack in-game.

"There is tremendous power in controlling how those sounds get triggered, how they are modified based on run-time data, player state, context, and much more. There is tremendous

[16] Collins, *Game Sound*, 86.
[17] Assets: in-game content (models, images, sound files, animations etc.).
[18] Collins, *Game Sound*, 112.

power in understanding all of that and designing content with the implementation context in mind."[19]

Middleware software development suites such as WWise, FMOD and iMus which allow for the manipulation of playback, attenuation and sound localization, give developers tools to connect audio assets to gameplay factors and are heavily used in audio production. Referring to work on the *Halo*[20] franchise, Collins notes composer Martin O'Donnell's claim that "…implementation is responsible for at least 50 percent of the final audio result."[21] By framing implementation in this way, O'Donnell highlights the interwoven agency at play in the development of a videogame's sound world.

Indeed, the creation of a videogame soundtrack involves several duties shared between composers, designers and engineers. The structure and scale of an audio team can vary greatly between projects and is often dependent on the budget of the game being developed. Collins lists the various roles and responsibilities of a videogame audio team as follows:

> **Sound Director(s)** (are) responsible for the overall audio vision and design of a title.
>
> **Sound Designers** will work with integrators and audio tool developers to create integrate and manage audio assets.
>
> **Dialogue/Voice-Over Artists** are, of course, responsible for providing the dialogue. They may be in-house, but it is more likely that they are cast externally and managed by the audio director.
>
> **Composers** are responsible for the music composition of the game. In smaller companies, they are frequently also responsible for the sound design.
>
> **Audio Programmers** or audio engineers are responsible for the audio tools, development and implementation of all audio assets in a game.

[19] Zdanowicz, Bambrick. *The Game Audio Strategy Guide*, XIV.
[20] Bungie. *Halo*: Combat Evolved. Microsoft Game Studios. Xbox. 2001.
[21] Collins, *Game Sound*, 112.

> **Licensing/Contracting Directors** are responsible for obtaining rights to licensed IP, and contracting out work externally.[22]

It is important to mention that the audio team can vary greatly in size and that each of these tasks are often managed entirely by a single person. In addition to the roles listed by Collins, there are further roles related to licensing and legal management of videogame music. While these roles don't directly impact creative aspects of soundtracks, they are nonetheless integral to the production and release of the final work.

Publisher

A *publisher* is generally the person or legal entity that owns the copyright to a work of art. These days, many independent musicians establish their own publishing companies to personally manage their intellectual properties to simplify the process.

Music Supervisor

A *music supervisor* is a person who oversees the choices of music brought into a project, such as a theatrical performance or a video game. They are expected to be competent in the legal side of music acquisition as well.[23]

Depending on the production schedule on a game, the audio team can vary greatly in size:

> [The] music team, for instance, consisted of four composers, three orchestrators, three ensembles (brass, string, and choir), a variety of ethnic soloists, and the development/ implementation team (Bajakian 2007). Smaller companies may have one or two people who must perform the equivalent of all of these jobs.[24]

Fundamentally, these roles break down to the simple categories of those involved in the development of sounds and those involved in the implementation of sounds. Tasks related to recording, licensing, or writing the music are inherently different to the processes of integrating

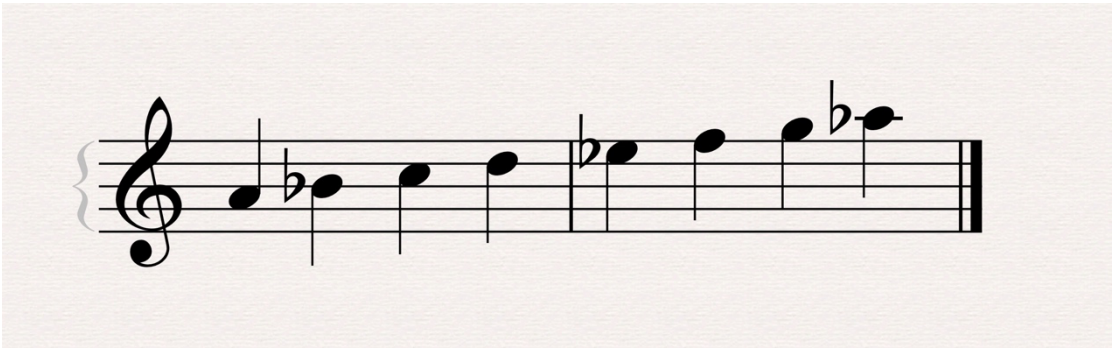
---

[22] Collins, *Game Sound*, 87.
[23]Robert Ciesla. *Sound and Music for Games : the Basics of Digital Audio for Video Games*. Berkeley, CA: Apress L. P., 2022.
[24] Collins, *Game Sound*, 89. For more detail, see Clint Bajakian, *Music: Design Production and Implementation,* Paper, presented at the annual Game Developers' Conference, San Francisco, March 4–9, (2007).

it, though both play an equal role in the final presentation of the soundtrack. This can be attributed to the non-linear presentation of audiovisual content in a game, which separates videogames from other media. Collins quotes Rouse's assertion: "Without non-linearity, game developers may as well be working on movies instead. (Rouse 2005 Chapter 7)."[25] Non-linearity in a videogame soundtrack is almost exclusively reliant on the implementation process. The various processes that trigger sound effects and music can be highly customised to maximise the impact of audio in a potentially infinite number of potential scenarios.

For example, the soundtrack to *Left 4 Dead,*[26] a multiplayer action-shooter where the player is pit against hordes of undead enemies, was created by composers Mike Morasky and Tim Larkin, who managed an audio team of sound designers and engineers to implement the music using a procedural AI system unique to the game. Morasky and Larkin use audio implementation to enhance the atmosphere of the game's narrative setting while informing the player on their immediate gameplay experience. The composers wrote each of *Left 4 Dead*'s cues based on a unique scale derived from the game's main theme (*Figure 6*).

Figure 6 – Morasky's Scale For Left 4 Dead



> [B]y using the singular and very chromatic scale, any piece written in it will generally dovetail pretty decently with any other. At the very least, the pieces seem to spring from the same musical universe. The resulting tunes are deceptively complex, easy to remember but difficult to sing accurately.[27]

[25] Richard Rouse, *Game Design Theory and Practice, second edition*, Plano, Texas: Wordware (2005), quoted in Karen Collins, *Game Sound: an Introduction to the History, Theory, and Practice of Video Game Music and Sound Design*, Cambridge, Massachusetts: MIT Press, 2008, 92.
[26] Valve South. *Left 4 Dead*. Valve. Microsoft Windows. 2008.
[27] *Developer Commentary (Left 4 Dead),* accessed 12 December 2022, https://left4dead.fandom.com/wiki/Developer_Commentary_(Left_4_Dead).

Modality in *Left 4 Dead* is used as a tonal rule-set, providing a consistent mood throughout the game. The chromaticism of the scale ensures that any dissonance is a controlled and deliberate compositional choice and ensures that the soundtrack has a coherent and recognisable thematic base, providing a means to build further pieces that fit with the rest. Morasky's choices here acknowledge the format his music is designed to fit. The soundtrack uses consistent modalities to ensure separate tracks can transition between each other with minimal conflict. *Left 4 Dead*'s soundtrack was implemented using a custom built 'Music Director' system, which tailored musical cues based on the experience of the player.

> We implemented a simple system to examine what's going on in the player's immediate environment, and add in the appropriate reactive scalar rule-sets to control the music and it's volume levels. Most of the more prominent musical cues are thus reactive results from rulesets processing this input, making the musical experience specific to each player.[28]

By using the Music Director system, *Left 4 Dead*'s soundtrack matches the intensity of each event presented in game, even if these events appear in a different order. This demonstrates that even if a soundtrack is composed to stylistically fit a game, it is the implementation process that ensures that the music is timed to fit appropriate in-game events. Morasky and Larkin's work is an example of composers making musical choices based on the implementation methods specific to a particular videogame. This treatment of implementation as an important process separate from composition is one which inspired my own soundtrack production process explored in Chapter 5.

1.3 Post-Production

Post-production concerns the finalisation of audio content presented within a game - mixing, mastering music, sound effects or dialogue, and correcting mistakes in the implementation process. Audio related duties in post-production may overlap with those in the production phase and vary from project to project. While the implementation process dictates the placement and playback of audio assets, the consistency or believability (e.g. tonal balance,

[28] *Developer Commentary (Left 4 Dead),* accessed 12 December 2022, https://left4dead.fandom.com/wiki/Developer_Commentary_(Left_4_Dead).

dynamic range, repetitiveness) of the implementation is fine-tuned in the post-production process. [29]

> The mixer must listen for what Charles Deenen calls *believability gaps* in the audio: awkward silences, too much repetition, an unnatural imbalance (in EQ, dynamic range, and so on).[30]

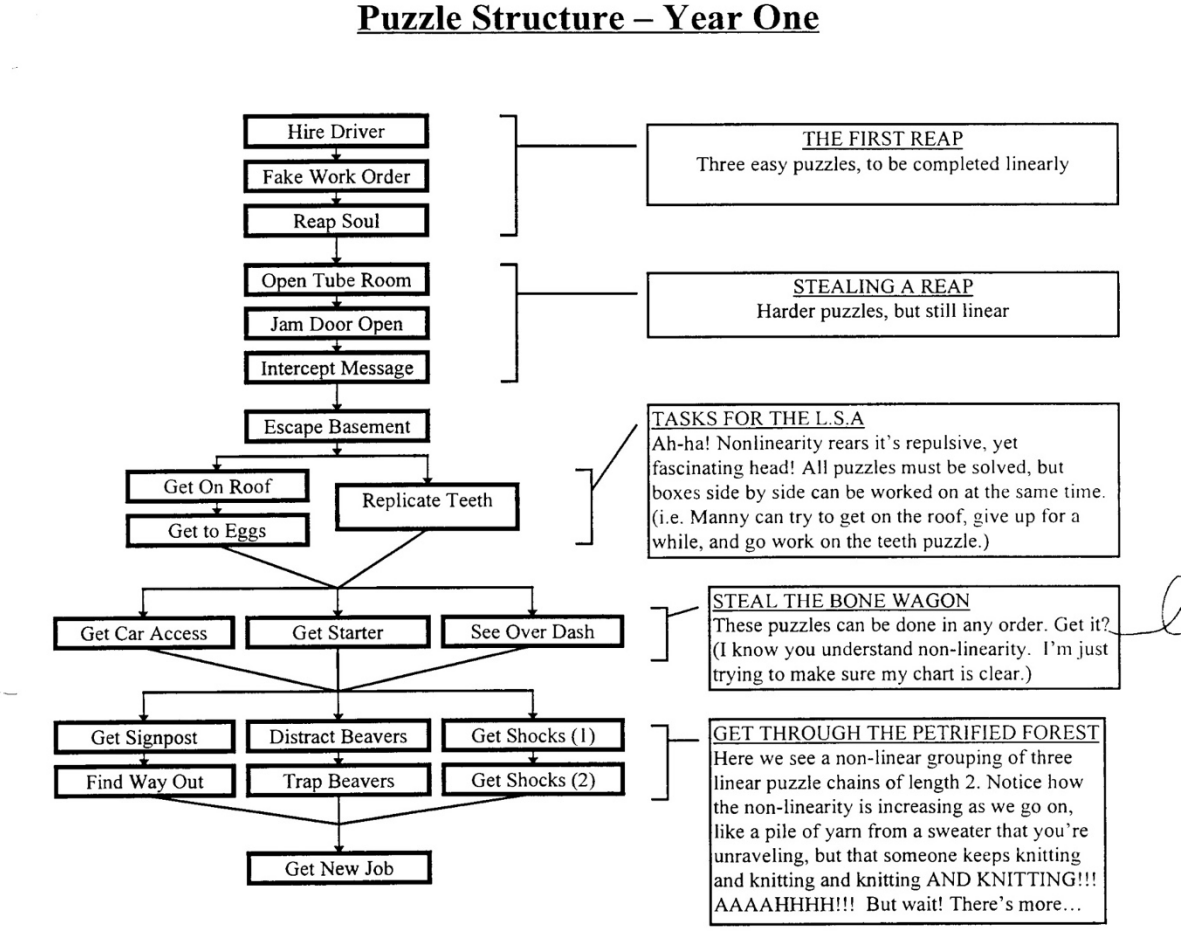
Post-production ensures that all audio assets fit within the sonic landscape presented by the game in a way that ensures the consistent attention and engagement of the player.[31] For instance, the reverb on a character's footsteps may change as they progress from an outdoor environment to an indoor environment. The original recording of the footsteps may be correct according to the design document and the timing of the transition between outdoor to indoor reverb may be synchronised, but the length or volume of the reverb might be incorrectly set. Games often feature hundreds of environments with different reverberation characteristics, with each needing to be checked and approved before the game can be commercially released. Considering the increasing capability and usage of audio implementation middleware, the importance of the post-production phase can be assessed by the sheer number of audio expressions that need reviewing.

1.4 Dynamic Range and Intensity During Development

From pre-production right through to post-production, development choices related to both composition and the implementation of music are disseminated *after* decisions related to intensity have been decided.[32] A flowchart depicting puzzle structure, for example, was one of the first images seen in the design document for Lucas Arts' 1998 puzzle game *Grim Fandango*[33] *(Figure 7)* with game director Tim Schaeffer specifically mentioning the increasing difficulty and number of puzzles taking place as gameplay advances.

[29] Collins, *Game Sound*, 112.
[30] Collins, *Game Sound*, 115.
[31] Collins, *Game Sound*, 115.
[32] Collins, *Game Sound*, 115.
[33] LucasArts. *Grim Fandango*. LucasArts. Windows. 1998.

Figure 7 - Fandago Puzzle Document – *Tim Schaeffer, Peter Tsacle, Eric Ingerson, Bret Mogilefsky, Peter Chan (1998)*



While there is no graph that shows the rising difficulty curve of the gameplay events, the structure of Peerdeman[34] and Lopez's[35] intensity arcs are present as events increase by complexity and difficulty as the story of the game progresses.

Discussion of a player's experience and the intensity arcs of videogames occur in each phase of game development, from pre-production to post-production. The various kinds of discussions outlined in this chapter represent standard industry practice in the development of a videogame soundtrack.

[34] Peerdeman, *Sound and Music*, 5.
[35] Lopez, *Gameplay Fundamentals Revisited.*

1.5 Intensity and the Player

Gameplay specific contexts can be more easily communicated to the player through audio which is synchronised with the rise and fall of gameplay intensity. As mentioned in the introduction, audio variation in challenge-based games such as *Pac-Man*[36] and *Tetris*[37] increases the sense of urgency within the player as the difficulty curve rises.[38] This relationship between intensity and gameplay has been shown to have a direct impact on performance during certain games, as found in the paper *Manipulating Player Performance via Music Tempo in Tetris*.

"Our results also show that the link between the music and the action of the game, i.e. *synchronism* of the tempo, is one of the criteria acting on the players' performance."[39]

The impact of soundtracks which are specifically tailored to gameplay events has been widely researched, particularly in the context of 'immersion',[40] where music and sound reinforces the player's awareness of spatial and narrative concepts from the virtual world. High levels of synchronisation between audio and visual material play a significant role in contributing to an immersive experience for the player, a concept explored throughout this thesis at multiple points. The following chapters establish that intensity arcs are already utilised currently as a function of game development and contribute to a player's experience as they engage with a game. In chapter two, I describe the artistic use of synchronised audiovisual content developed by sound designers to vary the intensity of action events in popular games. I unpack these events in relation to motifs of engagement in the field of videomusic and outline how these events help bind the player to a videogame's narrative arc. By dissecting the implementation of audiovisual synchronisation via intensity in popular games, I isolate goalposts for composers to aim their material when using intensity as a prompting device during development, further demonstrated in the execution of this concept as a framework in chapter four.

---

[36] Namco. *Pac-Man*. Midway. Arcade. 1980.
[37] Nintendo of America. *Tetris*. Nintendo of America. 1989.
[38] Rod Munday, "Music in Video Games," in *Music in the Digital Age*, ed. Jamie Sexton (Edinburgh: Edinburgh University Press, 2007), pp. 61.
[39] Hufschmitt, Aline, Stéphane Cardon, and Éric Jacopin. "Manipulating Player Performance via Music Tempo in Tetris." In *Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '20)*, 146-152. New York, NY, USA: Association for Computing Machinery, 2020.
[40] Munday, *Music in the Digital Age*, 53

Chapter 2

Videomusic and the Mapping of Audiovisual Units

Having established the role of intensity as a structural indicator and development aid, the next phase of this investigation is to pinpoint where measurements can be most helpful for composers. The intensity structures explored in Chapter 1 measure the impact of narrative events which are dependent on the believable link between auditory and visual phenomena.[41] Michel Chion highlights the creative potential of audiovisual intersection through the concept of synchresis:

> Synchresis (a word I have forged by combining synchronism and synthesis) is the spontaneous and irresistible weld produced between a particular auditory phenomenon and visual phenomenon when they occur at the same time.[42]

Sound and image assets in a videogame are created separately and are "welded" by developers to create the illusion of a virtual world. If the impact of sound in a videogame can be argued as resulting from creative use of synchresis, then a list of methods for using it can be used to add further descriptive capability to intensity measurements.

A typology of synchresis has been recently established in the field of videomusic, in which art works are charged by the designed relationship between sound and image. For example, Boucher and Piché[43] refine the language used by Chion[44] in his analysis of cinematic image and sound relationships to describe the various types of relationships that exist between audio and visual stimuli in videomusic. To clearly separate out and define each gesture for the purposes of analysis, Boucher and Piché segment works of videomusic into "audiovisual units":

---

[41] Collins, *Game Sound*, 115.
[42] Chion, *Audio-Vision,* 63.
[43] Myriam Boucher and Jean Piché, "Sound/image relations in videomusic" in *Sound and Image : Aesthetics and Practices*, edited by Andrew Knight-Hill (1st edition. Abingdon, Oxon ;: Routledge, 2020).
[44] Michel Chion, Walter Murch, and Claudia Gorbman, *Audio-Vision : Sound on Screen*, New York: Columbia University Press, 1994.

> An audiovisual unit (hereafter AVU) is a segment of variable duration (instantaneous, short, medium or long) that carries at least one perceptible sound/image relation that:
> - has a minimum of perceptual salience in the form of the work;
> - possesses well-defined kineto-morphological properties that make it discernable from other units in a given context;
> - plays a significant role in a larger discursive entity.[45]

Boucher and Piché identify ten types of synchresis which describe potential gestures in videomusic, categorised by the duration of their AVUs. Their resulting derivatives conveniently describe gestures which have permeated videogame design for decades. This gestural overlap provides a basis for applying Boucher and Piché's types of synchresis to videogame analysis. Furthermore, this overlap brings the potential to label the 'events' measured in Lopez and Peerdeman's graphs by their type of AVU. The following section compares Boucher and Piché's types of synchresis with examples from videogame play from 1980 to the present. These examples suggest creative goalposts which composers and sound designers may wish to align their resulting work.

2.1 Direct Synchresis

> Immediate synchronicity between a sound element and an image element. A hammer strikes a surface producing a sharp sound. The audio and visual relationship is bound by their simultaneity not by whether they are recognisable objects.[46]

Direct synchresis is the most frequent type of synchronised audiovisual feedback in games. Sound effects and musical cues are commonly played at moments of interaction to colour them by their degree of success. Namco's *Pac-Man*[47] uses direct synchresis between an animation and sound effect depicting the death of the player character. Nintendo's *The Legend of Zelda* franchise[48] directly correlates the animation of opening treasure chests to short music cues. Direct synchresis continues to play an important role in games today. CD Projekt Red's

[45]Boucher and Piché, *Sound/image relations in videomusic*, 17.
[46] Boucher and Piché, *Sound/image relations in videomusic*, 19.
[47] Namco. *Pac-Man*. Midway. Arcade. 1980.
[48] Nintendo EAD. *The Legend of Zelda*. Nintendo. 1986.

Cyberpunk 2077[49] and *The Witcher 3: Wild Hunt*[50] present science fiction and fantasy worlds respectively which require sound effects or music to be created by the developers rather than sourcing them from real world equivalents. Gunshots, explosions, menu interactions, alarms, alerts, and thousands more sound effects inform and educate the player on in-game parameters through direct synchresis, allowing the relationship between image and sound to appear believable to the player.

2.2 On-Cut Synchresis

> The simultaneous and abrupt change of sound and image in a simple straight cut edit. Cutting both image and sound at the same time. For example, the sound increases in volume and the image is cut at its peak amplitude.[51]

The intentional use of this effect is used in Naughty Dog's *The Last of Us,*[52] and Creative Assembly's *Alien: Isolation*[53] to heighten the impact of player death. Audio stingers, music, and dialogue are used exactly as described by Boucher and Piché, ramping up in volume and visual tension, cutting at the peak intensity of both.

2.3 Anchoring Synchresis

Two unrelated audio and visual objects remaining asynchronous for a time and then "falling together" into a state of sync which leads "to a sentiment of togetherness and the perception that audiovisual bonding is achieved."[54] The dynamic movement from asynchronicity to synchronicity in the soundtrack of *Guitar Hero*[55] is a good example of anchoring synchresis where the accuracy of a player's responses to prompts is conveyed by errors in the music.

---

[49] CD Projekt Red. *Cyberpunk 2077*. CD Projekt. PlayStation 4. 2020.
[50] CD Projekt Red. *The Witcher 3: Wild Hunt*. CD Projekt. PlayStation 4. 2015.
[51] Boucher and Piché, *Sound/image relations in videomusic*, 19.
[52] Naughty Dog. *The Last of Us*. Sony Computer Entertainment. PlayStation 3. 2013.
[53] Creative Assembly. *Alien: Isolation*. Sega. PlayStation 3. 2014.
[54] Boucher and Piché, *Sound/image relations in videomusic*, 20.
[55] Harmonix. *Guitar Hero*. Red Octane. PlayStation 2. 2005.

2.4 Expected Synchresis

> A significant event in the image while there is no change in the sound or there is a significant event in the sound while there is no change in the image.[56]

This technique is used extensively in videogame cutscenes and is inspired by similar usage in film. Activision's *Call of Duty: Modern Warfare 2* [57] depicts a nuclear explosion in space. The explosion is heard, and music signifies the horror of the event, but the debris and shockwaves fatally assaulting the player character and the space station around them are eerily silent. Similarly, Bohemia Interactive's *ARMA 3* [58] depicts nuclear explosions (at certain distances), as a bright light with no audio representation.

2.5 Delayed Synchresis

> An association of image and sound taking place at a deferred time. Delayed synchresis happens when concordance is re-established by an appropriate audio event that happens later and thus resolves the expectation of change.[59]

Instances of asynchronous sound and image in games are rare outside of a cinematic context (cutscenes). This is perhaps to avoid confusing the player who may think of the asynchronous moment as a bug rather than a feature inherent to the game. This effect is used as an auditory cue demonstrating the effect of a 'stun' grenade in multiple FPS action games such as *Titanfall 2*,[60] *Call of Duty 4: Modern Warfare*,[61] and *Onward*. [62] In the event of a grenade affecting the player character, it is common for a low-pass filter in tandem with a high-pitched ringing to

---

[56] Boucher and Piché, *Sound/image relations in videomusic*, 20.
[57] Infinity Ward. *Call of Duty: Modern Warfare 2*. Activision. Microsoft Windows. 2009.
[58] Bohemia Interactive. *Arma 3*. Bohemia Interactive. Microsoft Windows. 2013.
[59] Boucher and Piché, *Sound/image relations in videomusic*, 20.
[60] Respawn Entertainment. *Titanfall 2*. Electronic Arts. Windows. 2016.
[61] Infinity Ward. *Call of Duty 4: Modern Warfare*. Activision. Microsoft Windows. 2007.
[62] Downpour Interactive. *Onward*. Downpour Interactive. Windows. 2016.

overtake the existing soundscape until the effect clears and concordance is re-established. This can result in a lack of synchronisation between the visuals of the action around the player and the associated sounds: gunshots and explosions appear muted and slowly return to sonification.

2.6 Gestural Synchresis

> Forms on the screen move in a way that is congruent with the continuous sound and could believable be the source of the sound. Gestural synchresis happens when, directly or metaphorically, the movement in the image and in the sound follows the same "energy-motion" trajectory.[63]

Alien Isolation[64] utilises this technique in a sound effect and musical context. Sound effects illustrate the movement of the Alien as it stalks the player from unseen parts of the environment, providing contrasting spatial information intentionally confusing to the player. This creates the illusion that the player could be attacked from any direction at any time, as the direction of these sound effects do not directly correlate with any real action made by the Alien. To further this unease, the soundtrack also contains sparse instances of what appears to be diegetic sound made by the Alien, devoid of spatial information and sometimes played simultaneously with the aforementioned sound effects.

2.7 Ghost Synchresis

Ghost synchresis depicts "independence between image and sound components which make it possible to emphasise one element over the other."[65] An object at the forefront of the visual space without a corresponding audio object (or vice versa).

Bethesda's *Fallout 4* [66] uses the clicking of a Geiger counter to illustrate the invisible danger of nuclear radiation through sound effects. The surrounding sound effects, music and visual

[63] Boucher and Piché, *Sound/image relations in videomusic*, 20.
[64] Creative Assembly. *Alien: Isolation*. Sega. PlayStation 3. 2014.
[65] Boucher and Piché, *Sound/image relations in videomusic*, 21.
[66] Bethesda Game Studios. *Fallout 4*. Bethesda Softworks. Microsoft Windows. 2015.

environment do not provide further cues. Bethesda's *Elder Scrolls IV: Oblivion*[67] and *Elder Scrolls V: Skyrim*[68] tend to play combat themed musical accompaniment when the player is spotted by an enemy, even if that enemy is not near or visible to the player. This often results in a heavy contrast between musical accompaniment and visual imagery, as the threat communicated by the energy of the music is not at all evident in the image. This is arguably not intended by the developer or composer, as the same music plays during moments of combat which are visually acknowledged.

2.8 Behavioural Synchresis

Sound and image relationships which have "strong morphological correspondences" without total accurate representation - "An image of flying birds does not need to have the sound of the same number of flapping wings."[69]

Games sometimes use non-verbal audio cues in place of recorded non-player character (NPC) dialogue. This allows for the game to feature extensive dialogue via text without the need for recorded voice actors. Rare's *Banjo-Kazooie*,[70] and Hello Games' *No Man's Sky* [71] use different variants of this technique to indicate verbal dialogue even though it is clearly not recorded speech.

2.9 Congruent Movement Synchresis

Sounds associated with the image but not directly correlated. "The image is not perceived as the source of the sound but is associated with the sound by its comparable drive or momentum."[72] A technique used in a sound-effect context to add depth and believability to an in-game environment, even in circumstances where no virtual object actually causes the

[67] Bethesda Game Studios. *The Elder Scrolls IV: Oblivion.* Bethesda Softworks. Windows. 2006.
[68] Bethesda Game Studios. *The Elder Scrolls V: Skyrim*. Bethesda Softworks. Windows. 2011.
[69] Boucher and Piché, *Sound/image relations in videomusic*, 21.
[70] Rare. *Banjo-Kazooie*. Nintendo. Nintendo 64. 1998.
[71] Hello Games. *No Man's Sky*. Hello Games. Playstation 4. 2016.
[72] Boucher and Piché, *Sound/image relations in videomusic*, 21.

sounds.[73] Science fiction action/adventure games such as those in the *Mass Effect* trilogy[74] or *Star Wars: Squadrons,*[75] feature many computer-laden environments filled with blinking lights. These environments include ambient sound effect loops filled with beeps, clicks, and other computer associated noises correlated with the environment but not caused by it, reflecting the behaviour of the environment without a direct source of the sound. Congruent movement synchresis enables Austin Wintory's *Journey*[76] soundtrack to highlight the kinetic energy of the gameplay through elongated, slowly paced compositional gestures. Similarly, this congruence enables Mick Gordon's rhythmic, high-intensity accompaniment to evoke ferocity and violence by matching the energy of *Doom*'s[77] gameplay.

2.10 Metric Synchresis

AVUs which align sound and image via an external ruleset. "Metric synchresis also concerns periodic (predictable) synchronisation between a sound and an image, based on pattern recognition."[78] This is the primary technique behind rhythm games such as *Beat Saber*[79] and *Guitar Hero,*[80] or rhythm puzzles in games such as *Grand Theft Auto: San Andreas.*[81] Players receive visual cues timed with the BPM and time signature of the accompanying music to time their inputs.

2.11 Implications

"In order to forge new ground in the analysis of game music, it seems that the terms of the interpretation of videogames must be renegotiated."[82]

---

[73] Boucher and Piché, *Sound/image relations in videomusic*, 21.
[74] BioWare. *Mass Effect.* Microsoft Game Studios. Xbox 360. 2007.
[75] Motive Studio. *Star Wars: Squadrons*. Electronic Arts. Microsoft Windows. 2020.
[76] Thatgamecompany. *Journey*. Sony Computer Entertainment. PlayStation 3. 2012.
[77] id Software. *Doom*. Bethesda Softworks. Windows. 2016.
[78] Boucher and Piché, *Sound/image relations in videomusic*, 21.
[79] Beat Games. *Beat Saber*. Beat Games. Microsoft Windows. 2019.
[80] Harmonix. *Guitar Hero*. Red Octane. PlayStation 2. 2005.
[81] Rockstar North. *Grand Theft Auto: San Andreas*. Rockstar Games. PlayStation 2. 2004.
[82] Zach Whalen. "Case Study: Film Music vs. Video-Game Music: The Case of Silent Hill." In *Music in the Digital Age*, Edinburgh: Edinburgh University Press, 2007, pp. 74-75.

Integrating Boucher and Piché's typology into the discussion of game music provides ample footing for analysts and creators to form more granular propositions for their work. AVUs provide a helpful descriptive base for labelling audiovisual intersections which contribute to gameplay intensity. The above list of AVUs unfurls a range of musical impacts composers might strive to develop as they participate in the game design process. Integrating these notions of synchresis into the development process at any stage of production may afford composers and sound designers another way of thinking through how the sound assets they create resonate with players.

In the next chapter, I propose a methodology for capturing the intensity of particular videogame scenes that can be layered into the mix as composers seek out additional compositional prompts. The aim of this method is to provide videogame composers an additional tool they might use to weave the different kinds of synchresis outlined above into larger scale coherent expressions.

Chapter 3

Mapping Scene Intensity for Compositional Response

Visual behaviour is present in some capacity amongst all videogames, implied by the inclusion of 'video' in the name. Similarly, interactivity separates games from their linear counterparts like film and print media by including the audience as a participant in the experience. These fundamental concepts are specific to videogames and adhere to the aforementioned intensity structures that pace them. They also play a prominent role in motivating the use of in-game sound. As demonstrated in the list of AVUs in the prior chapter, visual behaviour and input is almost always associated with a corresponding sound effect or music cue. Peerdeman and Lopez describe the intensity of gameplay and gameplay assets moving in tandem, suggesting that a trend in one will be reflected by the other.[83] This suggests that as gameplay intensity changes, AVUs inevitably change as well. If trends in intensity affect visual and interactive behaviour, then these factors can be used to estimate the intensity of sound assets before they are inserted into the game.

Non-music assets for a game can be used by the composer to inspire or inform the choices behind their compositions. The variation in development scheduling and structure, however, makes it difficult to label specific assets for use in this context. For instance, a playable demo would be a useful non-musical asset for a composer to use to indicate the energy or atmosphere of their compositions. Unfortunately, playable demos may not be available until the production or post-production phases of development[84] and demos could exist entirely without complete 3D models or art assets, using placeholders that do not reflect the final product. Considering this inconsistency, no single asset can be flagged as a reliable indicator for composers around which to write their music. Given the reliability of intensity as an overarching structural indicator[85] however, trends in intensity may be as useful to composers as completed assets for determining musical structure. Assessing visual behaviour and interactivity as indicators for intensity may benefit composers with even the smallest list of completed assets to compose around.

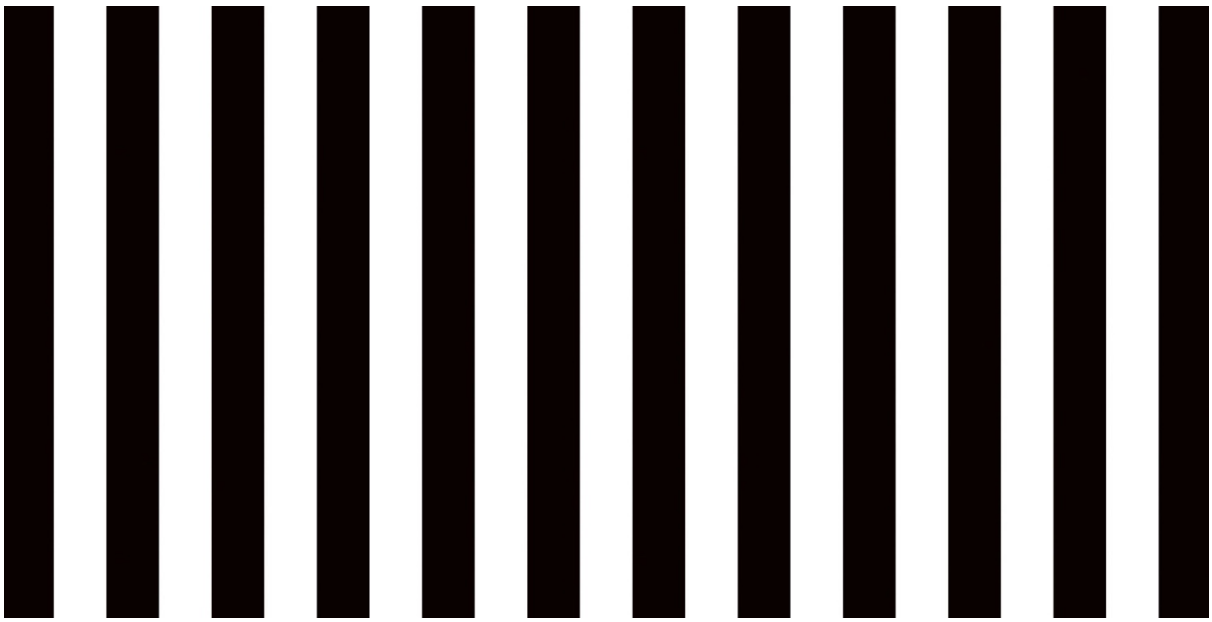
[83] Peerdeman, *Sound and Music*, 5; Lopez, *Gameplay Fundamentals Revisited*.
[84] Collins, *Game Sound,* 95-102.
[85] Peerdeman, *Sound and Music.*

In this chapter, I propose two metrics a composer might utilise in this context. A Pixel Change Ratio (PCR) presents the number of times a video sample moves between darkness and light. It indicates the speed of camera movement in the virtual space, gives a useful indication of visual intensity, and provides the composer with a variable against which they might synchronise accompanying musical content. An Actions Per-Minute (APM) measurement is a real-time indicator of the number of inputs and corresponding in-game reactions that a player has triggered in the last 60 seconds. This number is a direct measurement of player interactivity and can be used to determine the intensity of input required by a player.

3.1 Pixel Change Ratio (PCR)

A ratio of pixel changes measured during a sample of gameplay can help inform our understanding of gameplay intensity. This is a novel metric developed specifically for this research project. In *Figure 8*, a camera translating along the x-axis would capture images strobing from darkness to light at a rate determined by the speed of the camera's movement.

Figure 8 – Lighting Patterns In Videogames



The resulting footage will strobe (or flicker) at a higher frequency with faster camera movement and a lower frequency with slower camera movement. Visible content in a game creates the same effect as the camera moves. Player controlled camera movement is a deeply common function of 3D videogames. First-person narrative games such as *Call of Duty*,[86] *The*

[86] Infinity Ward. *Call of Duty*. Activision. Microsoft Windows. 2003.

Witness[87] and *Myst,*[88] and third person adventure games such as *The Last of Us,*[89] *Spyro,*[90] and *Fortnight,*[91] correlate camera movement with the right thumb-stick in console configurations, or mouse movement in PC configurations. Even in cases where the player has no direct control over the camera in games like *Resident Evil*[92] or *God of War*[93] higher rates of camera movement can still be aligned to similarly higher rates of interactivity, as engagement is a frequent trigger for camera transitions in these games. Given this tendency amongst modern day titles, the number of pixels sampled in this analysis could theoretically vary from 1 to 1000 and still present usable data to the analyst.

PCR is expressed as a ratio of the number of frames sampled over the frequency of the transition from dark to light. For example, in *Figure 8*, 91 frames sampled with a flicker rate - transition from dark to light or vice versa - of 13 would result in a PCR of 91:13 (ratios will remain non-simplified for ease of comparison). Greater ratios indicate a higher flicker rate and thus more on-screen behaviour, with lower ones presenting the opposite.

Figure 9 – A Typical FPS Screen Configuration (*Battlefield 1,* (DICE 2016))

[87] Thekla, Inc. *The Witness*. Thekla, Inc. PlayStation 4. 2016.
[88] Cyan, Inc. *Myst*. Broderbund. 1993.
[89] Naughty Dog. *The Last of Us*. Sony Computer Entertainment. PlayStation 3. 2013.
[90] Insomniac Games. *Spyro the Dragon*. Sony Computer Entertainment Games. PlayStation. 1998.
[91] Epic Games. *Fortnite: Battle Royale*. Epic Games. PC. 2017.
[92] Capcom. *Resident Evil*. Capcom. PlayStation. 1996.
[93] Santa Monica Studio. *God of War*. Sony Computer Entertainment. PlayStation 2. 2005.

Figure 9 indicates a configuration of on-screen elements typical of modern first-person shooters. A pair of arms hold a gun centre-right of the screen with accompanying GUI elements (map, crosshair, and ammo). Areas of the screen labelled A and B are the locations deemed suitable for PCR samples.

A) The centre of the screen, and the area directly impacted by the player's camera related inputs. The 'shooter' part of First-Person Shooter indicates that the player must align the crosshair in the centre of the screen to whatever they intend to shoot. As this is the primary method of expression within this genre of gameplay, a PCR sample here will detect changes in contrast most likely to be viewed by the player.
B) The area to the bottom left of the player. This area is less likely to be consistently viewed by the player, but more directly conveys the speed at which the virtual camera moves within the projected virtual space. This sample offsets problems with solely analysing the centre region of the screen, which are discussed further in *Figure 10.*

Figure 10 **-** *Race The Sun* (Flippfly 2013)

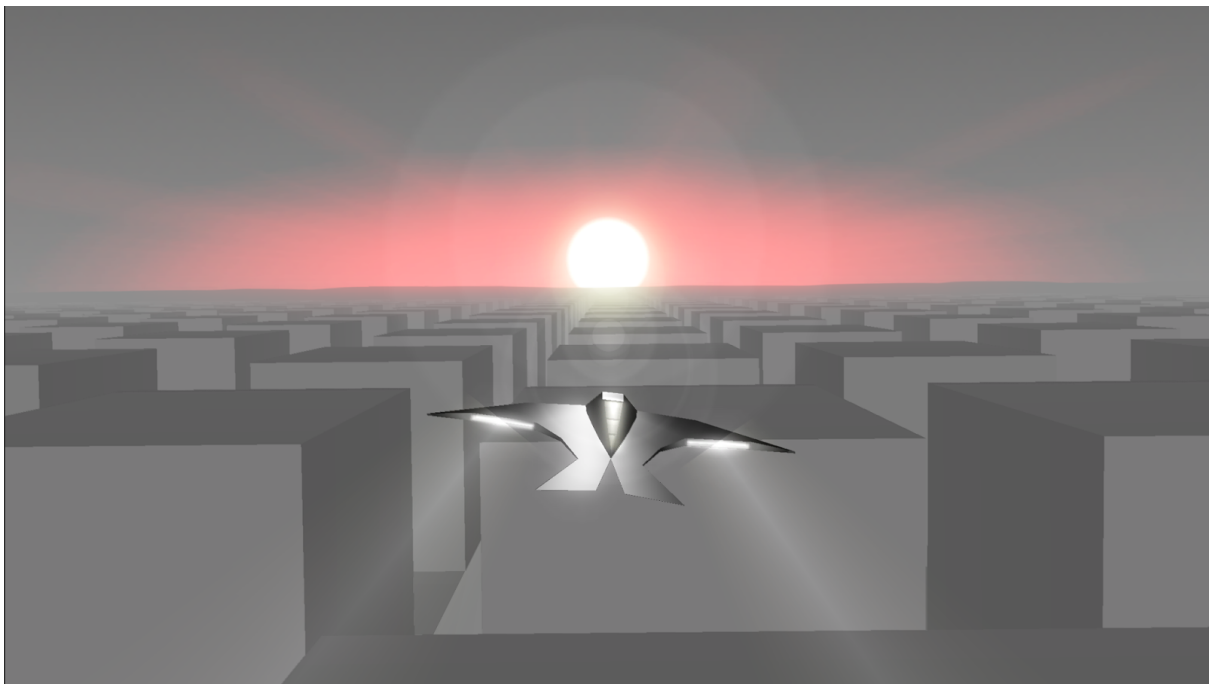


Figure 10 is a screenshot of *Race the Sun*,[94] which depicts a procedurally generated landscape to be traversed by the player in a forward direction towards the setting sun. Observable PCR data in the exact centre of the screen will be less useful due to the size and location of the sun,

[94] Flippfly. *Race the Sun*. Flippfly. Microsoft Windows. 2013.

and fewer pixels in this area of the screen will change based on the forward speed of the player, as the sun remains static as the player moves. To offset the 'blind spot' in the primary PCR sample, a second sample placed at a location on the outer edge of the observable screen would provide more useful data related to player and camera speed.

3.2 Actions Per-Minute (APM)

APM can be used as a tool for composers to determine how much attention and interactivity is expected from a player in any given section of gameplay. Actions Per-Minute is a reading of input related gestures made by the player in a finite space of time. It is a metric utilised for measuring efficiency in gaming and non-gaming applications alike. In the context of intensity analysis, APM provides readings of in-game player activity that may counterpoint or reinforce measurements of on-screen behaviour provided by PCR samples. This usage for APM has not been sourced from external research and faces certain limitations as a tool for effective intensity analysis. Analogue (or axis) based inputs which are made through mouse movements, thumb-stick or trigger movements on controllers complicate the means through which APM can be used as an indicator of engagement. Button inputs are relatively straightforward when compared to analogue inputs due to their binary nature: A single button press will often correspond to a single action. In contrast, analogue movement used for steering or camera rotation can be sporadic in some contexts and continuous in others. As a response to this limitation, the extent to which analogue inputs are used to measure gameplay intensity in this thesis will be limited to their presence and activity in the following demonstration. Analogue movements will be categorised as Not-Present, Sporadic, or Continuous. The relationship between player/camera movement and on-screen behaviour discussed in section 3.1 could indicate that PCR data could adequately cover the changes to analogue movement inputs not reflected by APM.

3.3 PCR/APM Demonstration

To demonstrate the potential application of these figures for the purposes of analysis, PCR samples were taken from three short recordings of gameplay from Motive's *Star Wars: Squadrons*.[95] This game was chosen for the demonstration due to its player movement and combat interactions which are common amongst a wide array of AAA[96] games. While gameplay is vehicle-motivated, player control is tied to the same configuration as first-person

[95] Motive Studio. Star Wars: Squadrons. Electronic Arts. Microsoft Windows. 2020.
[96] AAA: A classification of mid to high-budget games.

and third-person shooters in neighbouring genres, with player and camera movement linked to the left and right thumb-sticks respectively in console configurations, and through keyboard/mouse in PC configurations. Interactions between the player and NPCs (Non-Player Characters) are combat focused, requiring the player to align the centre crosshair with enemies before firing weapons: a very typical gameplay directive amongst vehicle-based and non-vehicle-based shooters such as the *Ace Combat* franchise,[97] or the *Halo* franchise[98]. Furthermore, *Star Wars: Squadrons* features an obstruction of the lower third of the screen by the cockpit of the player-vehicle, showing the need for an informed placement of PCR samples.

The demonstration in the following sections is designed to show the potential application of PCR and APM by comparing data taken from gameplay scenarios varied by intensity. The gameplay recorded was from a single-player portion of the game (Fleet Battles vs AI) which pits the player against NPCs in a competitive objective-based game-mode. The gameplay featured in the recording is easily replicated in other play sessions and is not unique to this demonstration. Three ten-second segments of footage were gathered from a recording 3:51 minutes in length, selected based on the intensity of the gameplay within. These were organised from estimated low-intensity to high-intensity and then analysed for PCR and APM data to confirm the estimates. The PCR analysis was improvised through a combination of screen-capture (OBS Studio) and video editing (iMovie) software, and APM was calculated using input data from the windows application 'DesktopAPM' in tandem with the GamepadViewer tool for OBS studio. Additionally, transitions from dark to light in each PCR sample were visually estimated rather than counting an exact average number of dark vs light pixels. These tools were used to demonstrate the potential application of PCR and APM measurements, which may be made more accurate with the development of more refined capture methods.

[97] Namco. *Air Combat*. Namco. PlayStation. 1995.
[98] Bungie. *Halo: Combat Evolved*. Microsoft Game Studios. Xbox. 2001.

Figure 11 – A Screenshot Of *Star Wars: Squadrons* Demonstrating PCR Sample Locations

Figure 11 displays the two areas where PCR samples were collected from the gameplay footage. The two square sections of the image labelled A and B were each sampled 91 times from the low, medium, and high-intensity sections of gameplay to demonstrate the changes in on-screen behaviour. These samples have been organised into grids for ease of comparison. The resultant number of images used in the PCR examples below presented usable data for demonstration purposes.

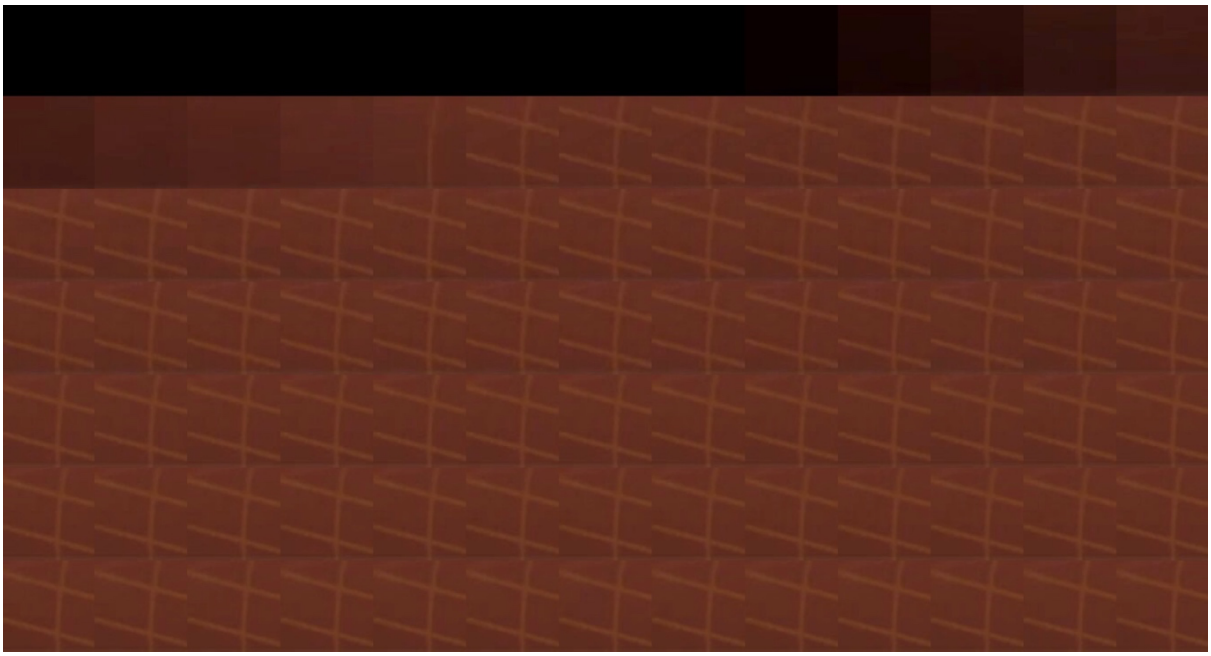
3.3.1 Low-Intensity Gameplay

This section features loading screens and menus interacted with prior to the flight and combat gameplay. Menus are a fundamental to a large majority of independent and AAA games, and commonly rely on formatted text and icons to organise and contextualise modes and available actions in the non-menu gameplay. On-screen behaviour and required input are generally lower as compared to non-menu gameplay, as demonstrated in the resultant PCR and APM analysis. Loading screens vary in number and duration, and often contain no interactivity outside of specific games like Ubisoft's *Assassins Creed* franchise,[99] or Bethesda's *Skyrim*[100] and

[99] Ubisoft Montreal. *Assassin's Creed*. Ubisoft. PlayStation 3. 2007.
[100] Bethesda Game Studios. *The Elder Scrolls V: Skyrim*. Bethesda Softworks. Windows. 2011.

Fallout[101] franchises. Despite this lack of input from the player, audiovisual presentations still feature in the loading screens of all games analysed in this dissertation. The presence of AVUs in these segments justifies their inclusion in the low-intensity gameplay analysis.

Figure 12 - Low-Intensity Gameplay Samples
Low-Intensity Gameplay: PCR Sample A



[101] Interplay Productions. *Fallout*. Interplay Productions. MS-DOS. 1997.

Low-Intensity Gameplay: PCR Sample B

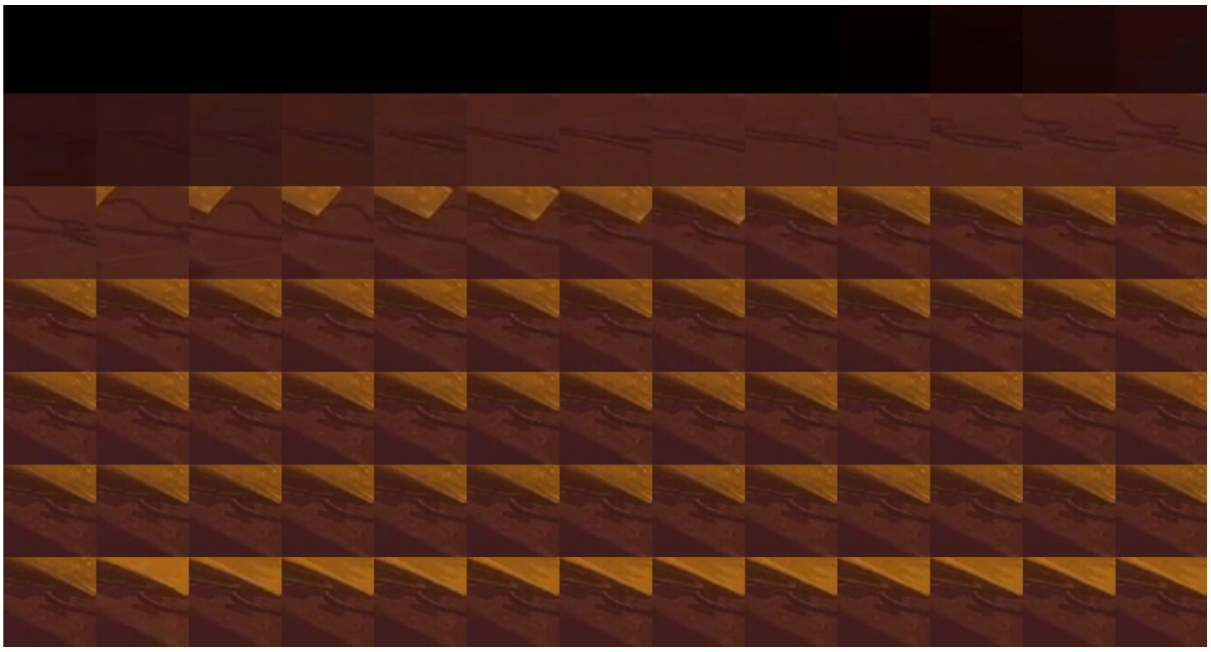
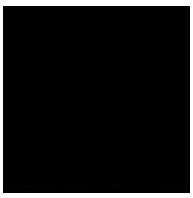


Figure 12 displays the PCR samples taken from the low-intensity section of gameplay. The primary activity in this section is missed by the samples due to their placement. No menu elements are visible, and the sampled sections are too small to indicate a holistic depiction of gameplay. However, the rate of strobing (as described in Section 3.1) is comparatively lower in both PCR samples of the low-intensity gameplay recording (*Figure 13*) when compared to the medium and high-intensity recordings. The sampled footage moves from its darkest displayed colours to its highest displayed colours only once:

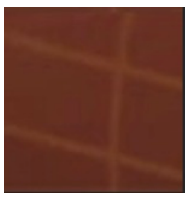
Figure 13 – Low-Intensity PCR Strobing

PCR A Strobing (91:1)

Darkest Colour

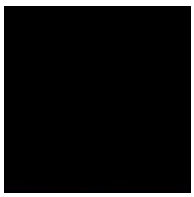
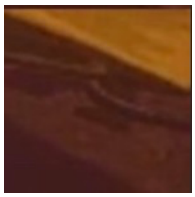


Lightest Colour



PCR B Strobing (91:1)

Darkest Colour Lightest Colour

This information provided by the samples highlights the tangibility of PCR analysis even in the face of low data yield. Furthermore, APM reinforces the hypothesis that this is a lower intensity section of gameplay. Six inputs were made by the player throughout the ten seconds of recorded footage, and no analogue inputs were present. The APM of this footage would remain at 6 as an isolated section of gameplay, but even a minute long sample with the same APM (36) would be lower than the other sections of footage. Each input measured here was accompanied by a sound effect played simultaneously: instantaneous AVUs correlating audiovisual stimulus with each accessed menu item.

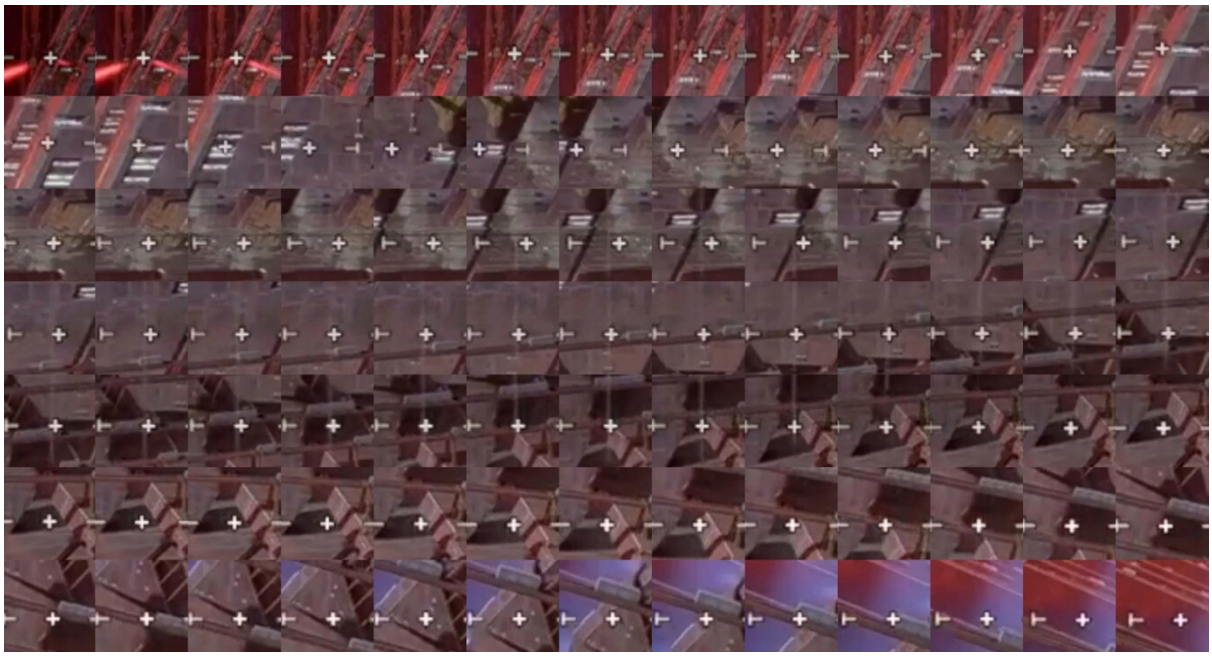
3.3.2 Medium-Intensity Gameplay

This section features the core flight gameplay exhibited by the game, but in a section at the beginning of the competitive game-mode where the player has not yet engaged enemy NPCs. This mirrors the lower level of player activity present in the early phases of non-videogame competitive activities such as chess, football, or poker, which precedes higher tension moments of interaction. The player is orienting themselves within the virtual environment with the knowledge that they may soon engage in-game complications. This sort of gameplay is present in open-world games such as *Grand Theft Auto: San Andreas*,[102] *The Witcher 3: Wild Hunt*,[103] and *Cyberpunk 2077*[104] where the player can explore outside of conflict. Inputs are primarily movement and camera related, with no offensive or defensive actions made.

Figure 14 – Medium-Intensity PCR Samples

[102] Rockstar North. *Grand Theft Auto: San Andreas*. Rockstar Games. PlayStation 2. 2004.
[103] CD Projekt Red. *The Witcher 3: Wild Hunt*. CD Projekt. PlayStation 4. 2015.
[104] CD Projekt Red. *Cyberpunk 2077*. CD Projekt. PlayStation 4. 2020.

Medium-Intensity Gameplay: PCR Sample A



Medium-Intensity Gameplay: PCR Sample B

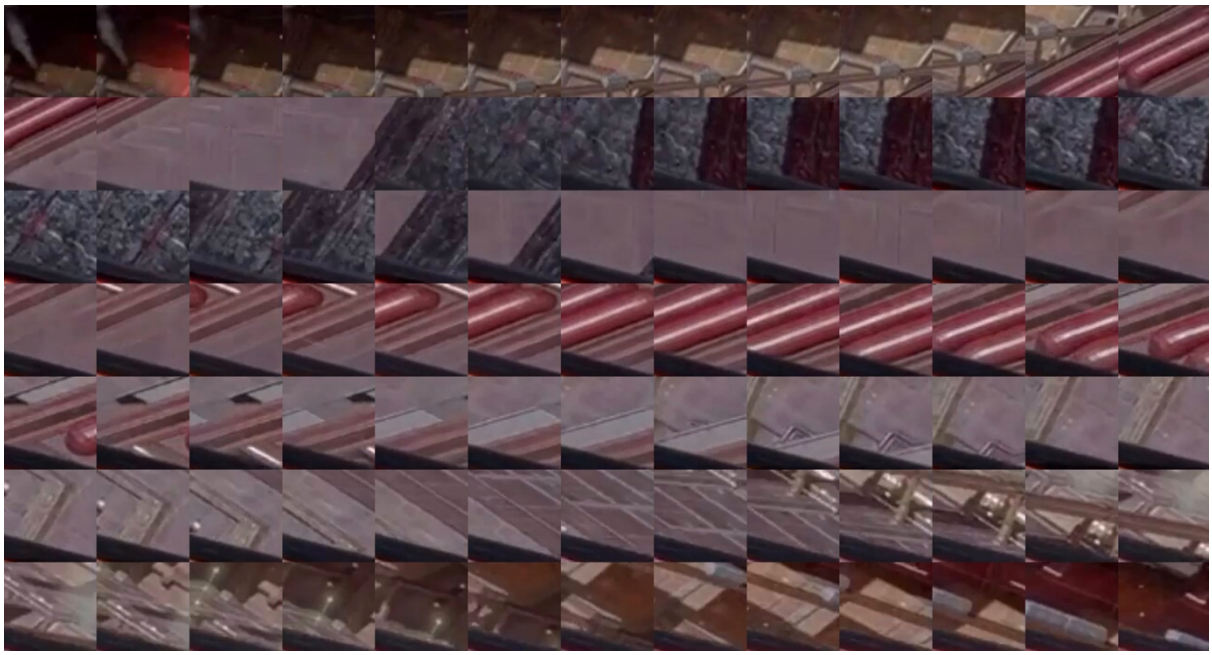


Figure 14 shows the PCR samples taken from the medium-intensity gameplay and indicates higher activity compared to the low-intensity samples in *Figure 12*. The strobing of the sampled images is higher, moving from dark to light multiple times over the sample. Each transition from one state to another is counted to inform the final ratio as displayed by the '#' symbols in *Figure 15.*

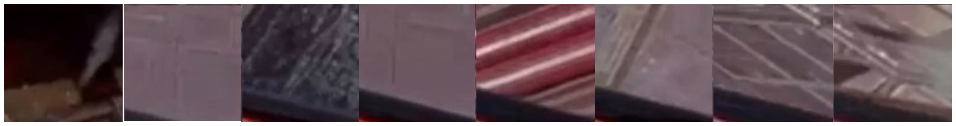
Figure 15 – Medium-Intensity PCR Strobing

PCR A Strobing (91:3)

Dark # Light # Dark # Light

PCR B Strobing (91:7)

Dark # Light # Dark # Light # Dark # Light # Dark # Light



The APM of this section is still low in comparison to the high-intensity recording but involves continuous analogue movement. Constant micro-adjustments to player movement and rotation are made throughout the entire duration of the recording, with no combat actions made. This

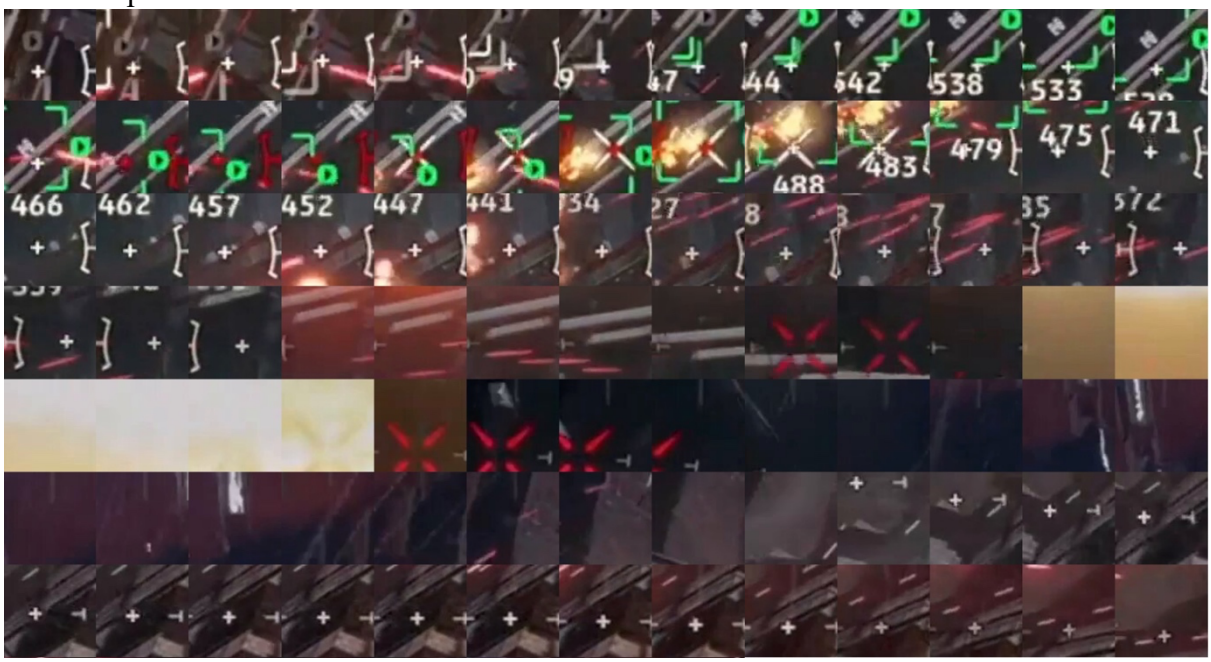
reflects a similar increase in gameplay intensity as suggested by the PCR data and reinforces that there has been a change in the required player-interactions.

3.3.3 High-Intensity Gameplay

The gameplay in this recording is combat and movement focused, with higher levels of player-engagement and on-screen behaviour. The player has engaged with enemy NPCs of different threat levels and is required to combine their knowledge of the virtual space and in-game potential behaviours to form competitive strategies in real-time.

Figure 16 - High-Intensity Gameplay PCR Samples

PCR Sample A



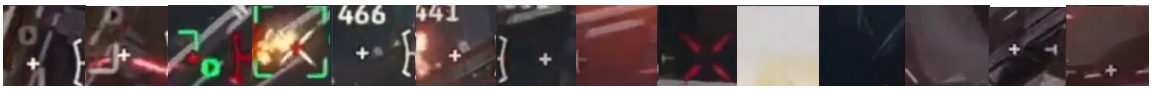
PCR Sample B

Figure 17 shows a significant increase to the rate of strobing present in the high-intensity PCR samples (*Figure 16*). Both samples move from dark to light almost continuously, creating a busier on-screen projection than the low and medium-intensity recordings. (Dark and Light are labelled “D” and “L” respectively for ease-of-formatting.) The resultant PCR ratio is the greatest of the three examples.

Figure 17 – High-Intensity PCR Strobing

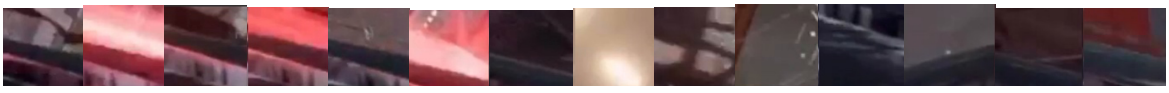
PCR A Strobing (91:13)

D L D L D L D L D L D L D L



PCR B Strobing (91:13)

D L D L D L D L D L D L D L



The APM of the high-intensity section of gameplay sits at 11 and analogue movement is continuous, which is the highest level of input activity recorded in the demonstration.

Movement and combat related inputs are intertwined as the player strategizes, suggesting a higher level of attention required.

3.3.4 Other Examples

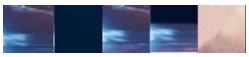
The same analysis was conducted on three other games to evaluate the trend of higher PCR/APM readings correlating with higher intensity gameplay (*Maneater*,[105] *The Witcher 3: Wild Hunt*,[106] and *Deathloop*[107]).

Figure 18 – Maneater Intensity Measurements*:*

Low-Intensity

(PCR A = 91:2 | PCR B = 91:4)

APM = 3 (No Analogue Movement)



Medium-Intensity

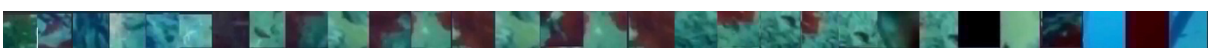
(PCR A = 91:6 | PCR B = 91:19)

APM 14 (Continuous Analogue Movement)

High-Intensity

(PCR A = 91:18 | PCR B = 91:29)

APM 26 (Continuous Analogue Movement)



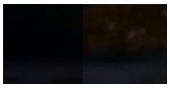
[105] Tripwire Interactive. *Maneater*. Tripwire Interactive. PlayStation 4. 2020.
[106] CD Projekt Red. *The Witcher 3: Wild Hunt*. CD Projekt. PlayStation 4. 2015.
[107] Arkane Lyon. *Deathloop*. Bethesda Softworks. PlayStation 5. 2021.

Figure 19 **-** *The Witcher 3: Wild Hunt* Intensity Measurements

Low-Intensity

(PCR A = 91:1 | PCR B = 91:1)

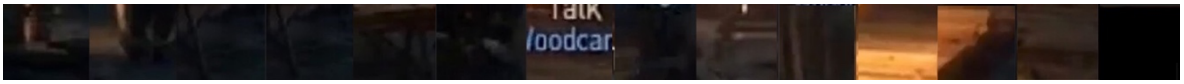
APM 5 (No Analogue Movement)

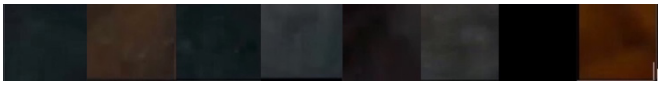


Medium-Intensity

(PCR A = 91:13 | PCR B = 91:7)

APM 2 (Continuous Analogue Movement)

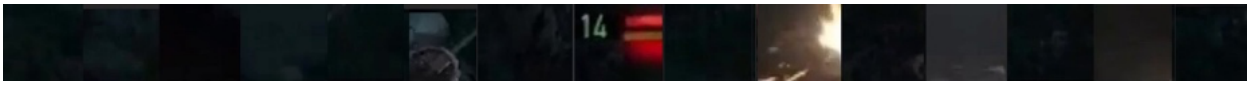




High-Intensity

(PCR A = 91:14 | PCR B = 91:8)

APM 10 (Sporadic Analogue Movement)



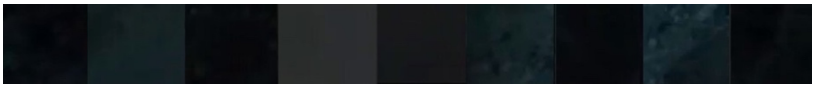
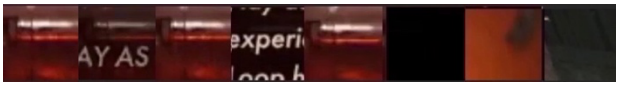


Figure 20 – Deathloop Intensity Measurements

Low-Intensity

(PCR A = 91:7 | PCR B = 91:2)

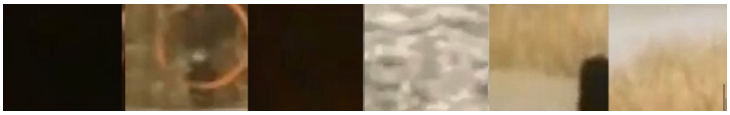
APM 2 (No Analogue Movement)

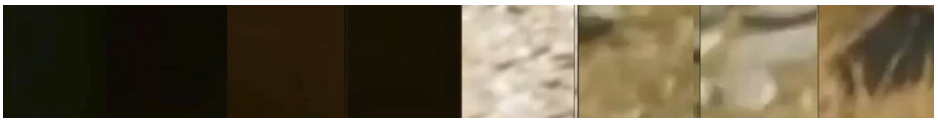


Medium-Intensity

(PCR A = 91:13 | PCR B = 91:7)

APM 1 (Continuous Analogue Movement)

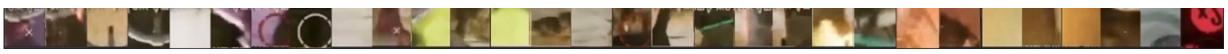




High-Intensity

(PCR A = 91:14 | PCR B = 91:8)

APM 13 (Sporadic Analogue Movement)



3.4 Implications for Composition

Despite the strictly visual nature of PCR analysis, it produces a wealth of variables for use in developing music or sound effects. Typically, adaptive soundtracks have relied on transitions between gameplay sections or the presence of in-game variables such as enemies, puzzles or powerups to trigger music or sounds. Tying a sound to an on-screen object is simple to implement and widely utilised, as seen in the prior examples of synchresis discussed in Chapter 2. The creation of a realtime PCR analysis tool could provide a more granular reading of on-screen changes in games and thus a basis for more nuanced sonic triggers. Realtime APM tools already exist but haven't been elevated as tools for audio implementation.

In the context of composition, PCR is a highly precise means of performing a simple observation of on-screen behaviour. Theoretically, a composer could assess changes to on-screen behaviour by simply looking at areas of the screen appropriate for PCR samples and estimating their flicker rate as low, medium or high. APM is a similarly complex method of transcribing the easily observed number of buttons the player has pressed recently. Furthermore, waveforms from the audio accompanying *Maneater*, *The Witcher 3*, and *Deathloop* reflected the different levels of intensity observed by the PCR/APM analysis through amplitude. In fact, *The Witcher 3* is the only instance of APM data being higher in the low-intensity analysis compared to the medium-intensity section (*Figure 19*), a contradiction matched by the waveform amplitude in each of those sections (*Figure 22)*. This indicates that the artists behind the audio of the analysed games are already utilising indicators which share similar data to PCR/APM analysis, and that these figures could be used to formalise techniques already circulating within the games industry.

Figure 21 - Maneater Waveforms (Low/Medium/High-Intensity)

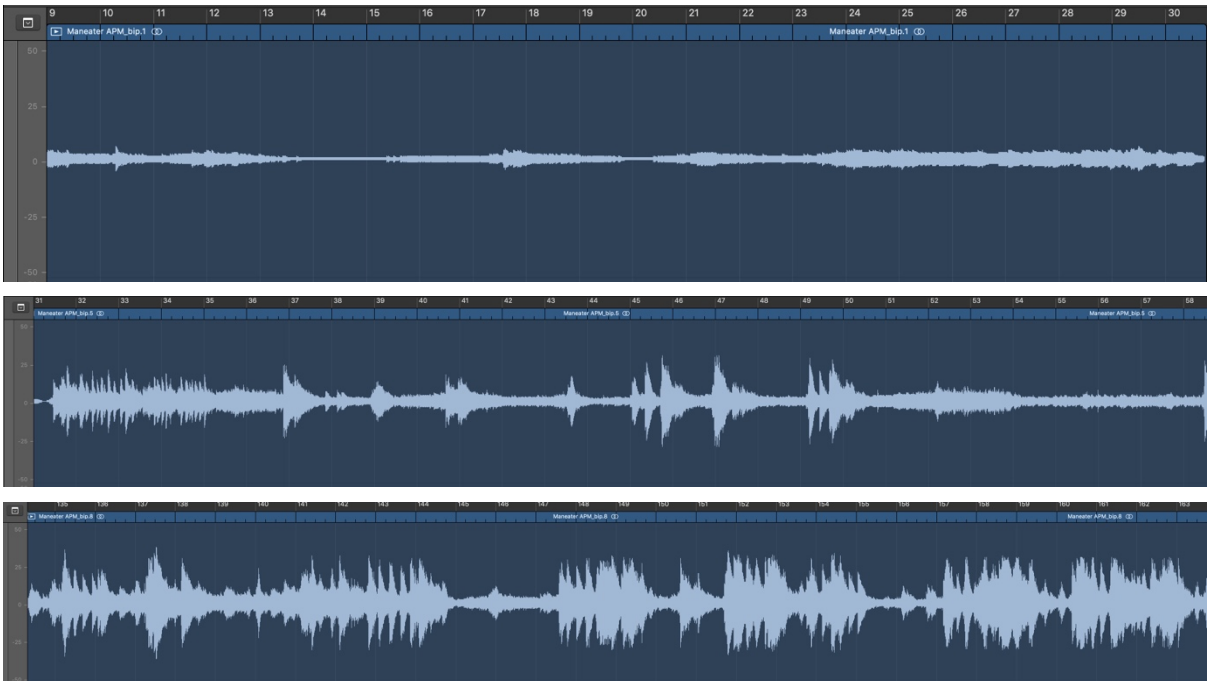


Figure 22 - The Witcher 3 Waveforms (Low/Medium/High-Intensity)

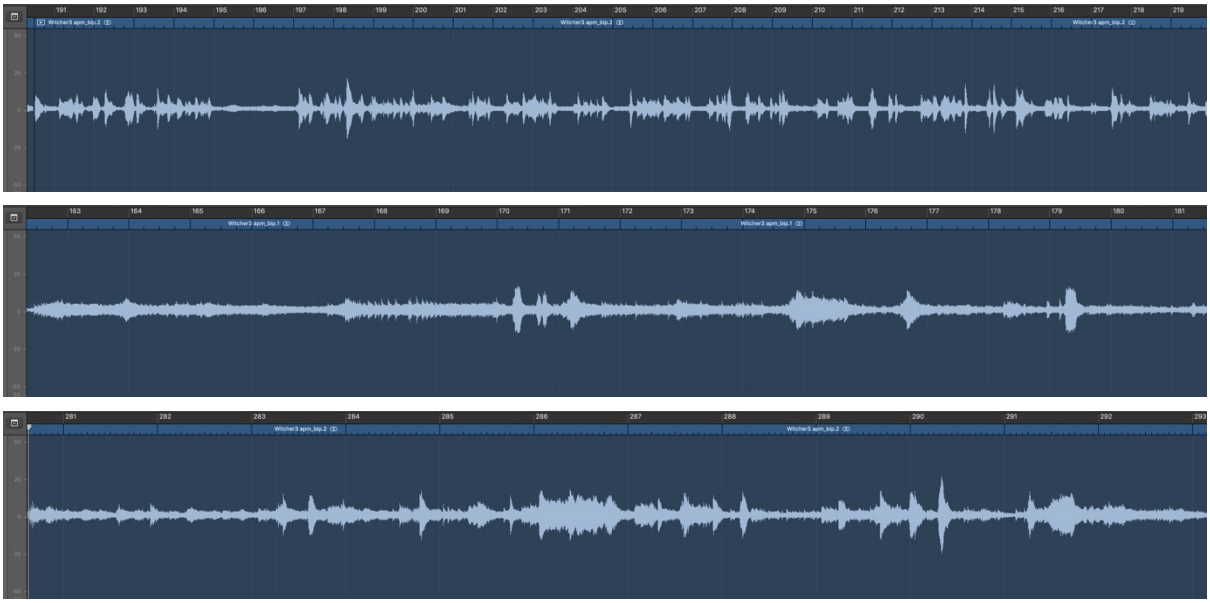
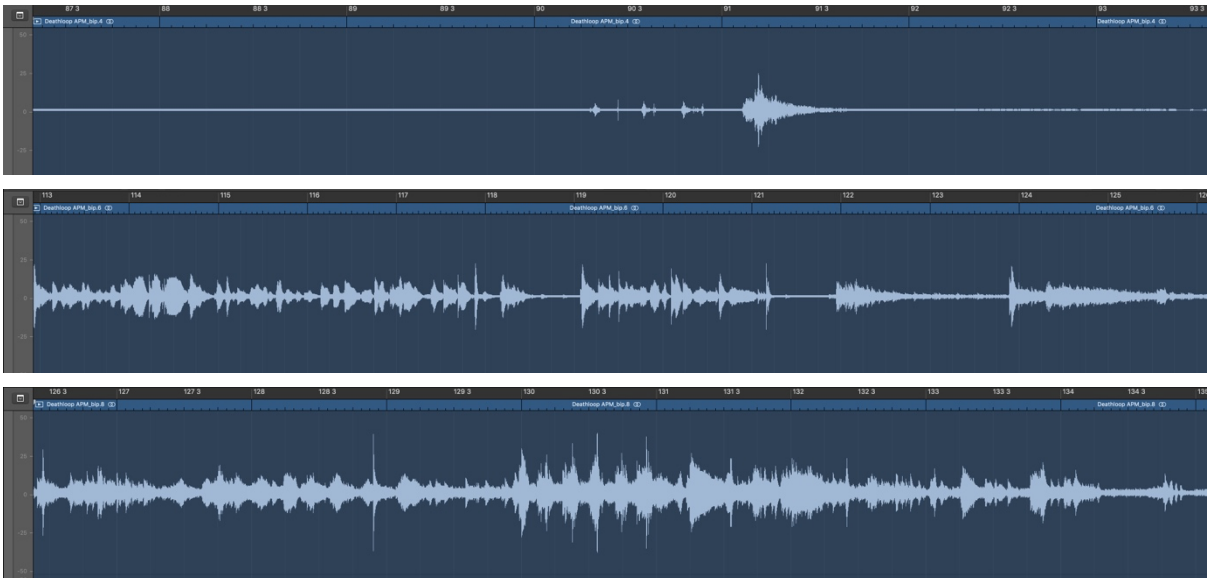


Figure 23 - Deathloop Waveforms (Low/Medium/High-Intensity)



3.5 PCR/APM Limitations

This evaluation of PCR and APM is a proof-of-concept to highlight possible uses for videogame development. There are three limitations to APM and PCR which must be addressed before they can be considered reliable, accurate figures in the context of measuring intensity.

3.5.1 Data Accuracy

Gameplay tropes and other commonalities between videogames play a large role in the reliability of these figures in predicting a player's relationship with games. The examples demonstrating these figures for analysis purposes are all 3D action games from the last 10 years set in a semi-open world environment. However, the accuracy through which PCR and APM illustrate higher or lower levels of required engagement is reliant on the analyst's ability to determine the average values of both in a given section of gameplay. The games analysed in the demonstration were chosen with prior knowledge of their average input requirements, which informed their intensity labels. Even though the demonstration appears to confirm APM and PCR can measure two different kinds of intensities across multiple titles, more specific collection methods may need to be developed before these measurements can be considered accurate.

PCR may remain entirely consistent in low and high-intensity sections within games which present limited changes to visual behaviour such as *Zork.*[108] Some games, such as *Inscryption,*[109] present high-intensity scenarios which may cause the player to sit in complete silence as they ponder their next move. A scenario like this could be considered one of high intensity despite low PCR and APM readings. In the context of 'speed-runs' (a subsection of competitive gaming which prioritises completion speed), players will utilise unconventional combinations of in-game behaviour to minimise time spent moving from section to section. Even outside of a competitive format, games such as *Maneater*,[110] *Blade & Sorcery*,[111] or *Vampyr*[112] which allow the player to move in a forward momentum at a faster rate through jumping or dodge mechanics may feature higher APM/PCR data in sections of relative low-intensity as players favour the faster movement. This may imply the need for contextual input analysis when using these metrics for music triggers. Furthermore, tools to determine the exact number of pixels which transition between dark and light during a PCR sample may also require developing before the rate of strobing can be considered a reliable measurement.

3.5.2 PCR Sample Placement

The placement or number of PCR samples may need to vary from game to game based on elements which may obscure parts of the screen like maps, tutorial text, or health-bars. User interface elements like these often do not reflect dynamic lights in the 3D space and sampled pixels here may appear static even during changes in gameplay intensity. Additionally, first person vehicle-based games like *Ace Combat 7*[113] or *Forza: Horizon*[114] frequently cover the entire bottom third of visible screen space with vehicle interior. These obstruction issues could prevent PCR samples from accurately measuring player speed unless manually placed at points where the surrounding environment is most visible.

---

[108] Infocom. *Zork*. Personal Software. PDP-10. 1977.
[109] Daniel Mullins Games. *Inscryption*. Devolver Digital. Windows. 2021.
[110] Tripwire Interactive. *Maneater*. Tripwire Interactive. PlayStation 4. 2020.
[111] WarpFrog. *Blade and Sorcery*. WarpFrog. Microsoft Windows. 2018.
[112] Dotnod Entertainment. *Vampyr*. Focus Home Interactive. Microsoft Windows. 2018.
[113] Bandai Namco Studios. *Ace Combat 7: Skies Unknown*. Bandai Namco Entertainment. PlayStation 4. 2019.
[114] Playground Games. *Forza Horizon*. Microsoft Studios. Xbox 360. 2012.

3.5.3 PCR Conversion and Formatting

The number of samples in the recorded footage was set arbitrarily by the "MP4 to JPG Converter" tool from image.online-convert.com, which was used to save PCR sample recordings as images. The total length of footage converted for each game averaged to 30 seconds, providing 300-400 frames each. These were then divided into three groups based on their corresponding sections (low, medium, high-intensity), and aligned into grids of 91 samples using Microsoft Word. More specialised image editing applications were evaluated (GIMP, *Adobe Photoshop CC*), but deemed unsuitable due to slow-acting image alignment tools. Repeated attempts at capturing PCR data using this method returned a flicker rate which increased alongside gameplay intensity in all four analysed games and therefore was deemed suitable for the demonstration.

3.6 Summary of Findings

By creating a direct link between gameplay intensity and the aforementioned metrics, real-time information can be derived from any number of gameplay demos or videos for the purposes of compositional prompts. By contextualising PCR and/or APM measurements within the typology provided in Chapter 2, composers have a potential vector for planning and executing specific artistic intentions through direct indicators of gameplay intensity. By combining this approach with existing methodologies and practices in the industry, composers could maximise the impact of their music and the cohesion of the interactive experience for the player. Further research of PCR and APM measurements could create the foundation for software which can identify changes between low and high intensity in real-time, providing statistical information for usage in a development or even post-release context. The potential benefits posed by these measurements can only be determined by their application to real videogames to avoid the concepts proposed by this chapter remaining purely speculative. To that end, PCR and APM measurements were organised into the Gameplay Intensity Compositional Framework discussed in Chapter 4 to demonstrate intensity measurements as a tool for composition and sound design in a real-world context.

Chapter 4

Gameplay Intensity Compositional Framework

This framework was developed in the pursuit of a method to quickly compose a soundtrack for a videogame with limited planning and few completed assets. It draws together the traditional compositional prompts utilised in videogame development discussed in Chapters 1 and 2 with the novel prompts outlined in Chapter 3. In Chapter 5, I discuss in detail how this framework was used in the development of the major creative work submitted in this thesis, the soundtrack for the game *Unsteady VR*.

4.1 Framework

The Gameplay Intensity Framework uses interactivity and visual prompts to determine shifts in intensity in sections of gameplay of varying length. Measurements of gameplay might be of the full game, individual levels, or short segments within a level. The purpose of performing a measurement is to determine what constitutes the scene being scored – i.e. the narrative arc of the action that is being set to music, so that a composer might get a sense of the musical drama required.

Figure 24 – Intensity And Pacing Plan Example (Lopez, 2008)

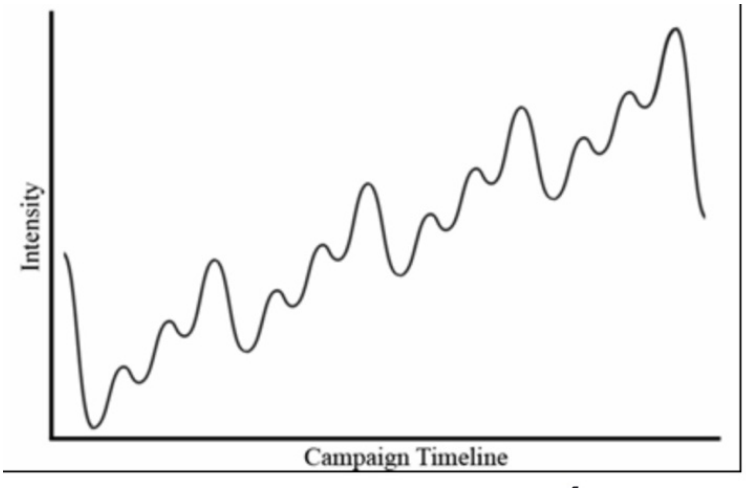
Initial Intensity & Pacing Plan Example

Location	Key Action Event	Key Event Description	Ideal Intensity Rating Targets	Projected Intensity Rating
Mega Yacht	Prologue Escape	Short Description	8	4
Wharf	Falling Crane	Short Description	1	2 (+1)
	Exploding Tank	Short Description	2	2
	Crumbling Catwalk	Short Description	3	5 (+2)
Skyscraper	Scaffolding Chase	Short Description	2	2
	Elevator Shaft Descent	Short Description	3	4 (+1)
	Helicopter Attack	Short Description	4	4
Sewer System	Flood Rapids Chase	Short Description	3	2 (-1)
	Giant Waterfall	Short Description	4	4
	Crumbling Ceiling	Short Description	5	5
Factory	Liquid Coolant Accident	Short Description	4	5 (+1)
	Transformer Shortout	Short Description	5	5
	Assembly Line Implosion	Short Description	6	9 (+3)
Freeway	18-wheeler Hijack	Short Description	5	5
	Car Chase / Shootout	Short Description	6	6
	Gas Tanker Explosion	Short Description	7	8 (+1)
Oil Refinery	Infiltration Under Alarm	Short Description	6	6
	Catwalk Chase	Short Description	7	7
	Cascading Explosions	Short Description	8	9 (+1)
Private Island	Alerted Villa Assault	Short Description	7	6 (-1)
		Short Description	8	9 (+1)
		Short Description	9	9
...	...	...	...	...

4.1.1 Analysing Visual Behaviour and Interactivity

Lopez' graphs indicate the usefulness of intensity planning for game design, even in the case where intensity is purely estimated.[115] Each level is rated based on an estimated level of intensity on a table (*Figure 24*) and then organised into a line graph (*Figure 25*). The Gameplay Intensity Framework attempts to fulfil the same purpose, but by using qualities from the game itself.

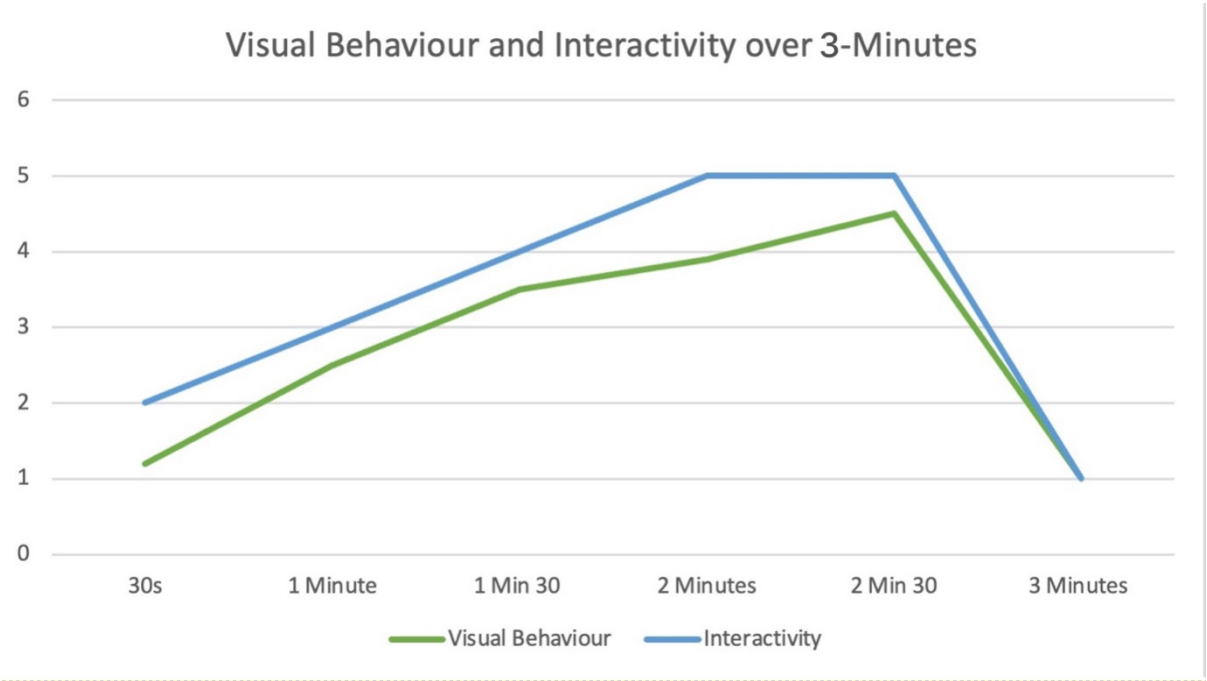
Figure 25 – A Generic Campaign Timeline (Lopez, 2008)



For example, a composer is provided with a 3-minute segment of gameplay footage as a reference for composing music. By looking for changes to visual behaviour and interactivity, the composer can pinpoint fluctuations to intensity in both categories as gameplay progresses (*Figure 26*). This can be achieved through estimation or direct measurement using APM and PCR.

[115] Lopez, *Gameplay Fundamentals.*

Figure 26 – Visual Behaviour And Interactivity Graph

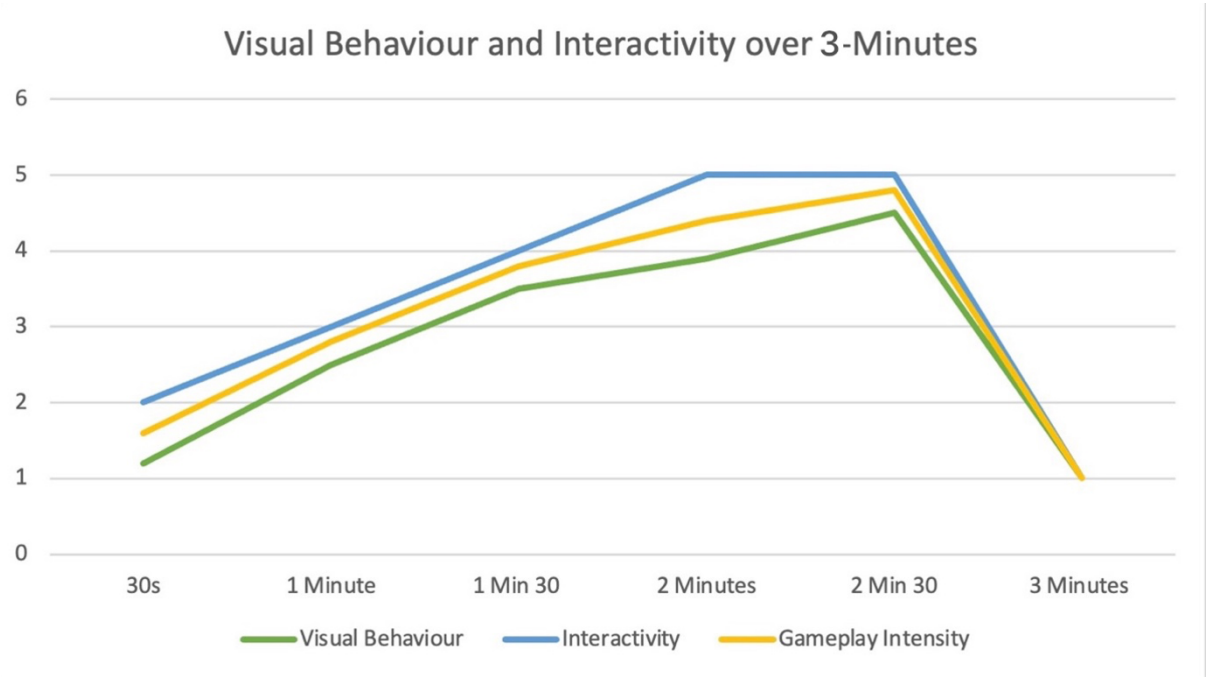


Comparing the ratio of pixel changes every 30 seconds will provide a direct measurement of visual behaviour for analysis. APM can be graphed in the same way, with higher APM indicating a higher requirement of player input. These direct measurements are useful for games which constantly present high levels of both visual and interactive behaviour, in which tiny changes in intensity are harder to witness with the human eye.

4.1.2 Measuring Intensity

Provided the graphed data shows a shared trend across both visual and interactive behaviour, the composer can more easily confirm the movement of gameplay intensity. In circumstances where contradictory data is presented, the composer can analyse additional stimulus to uncover overarching intensity trends (*Figure 27*). Analysing the footage for the number of instantaneous AVUs over time, for example, will directly reveal how many moments of synchresis occur over time. Gunshots, melee attacks, player-received damage or objects collected by player all make noises and will increase or decrease with gameplay intensity. A correlation between these trends and either visual behaviour or interactivity will provide the composer with enough data to estimate gameplay intensity. The value provided by this approach comes from sourcing intensity directly from gameplay elements rather than conceptual story beats which aren't directly connected to the second-to-second gameplay experienced by the player.

Figure 27 – Gameplay Intensity Added To Visual Behaviour And Interactivity Graph



4.1.3 Applying Intensity Measurements

After a consistent sample of gameplay intensity has been gathered, the composer can determine the appropriate musical gestures for the scene. The sonification of intensity changes may be implemented directly, with increases in intensity to higher amplitude, texture, or changes to instrumentation. Lower intensity may be reflected by more ambient tracks, directly linking the energy of the scene to a suite of sparse melodic gestures. Boucher and Piché's[116] typology in Chapter 2 provides a number of sites/relationships that may be specifically targeted during this process with moments of differing gameplay intensity changing the number and perhaps nature of AVUs present in a scene. Regardless of whether a composer has directly counted the number of AVUs, rises and falls in intensity will counterpoint the dialogue and sound effects embedded within a particular scene. As discussed in Chapter 2, the totality of this sound world aims to bind the player more fully into the game's overall narrative arc.

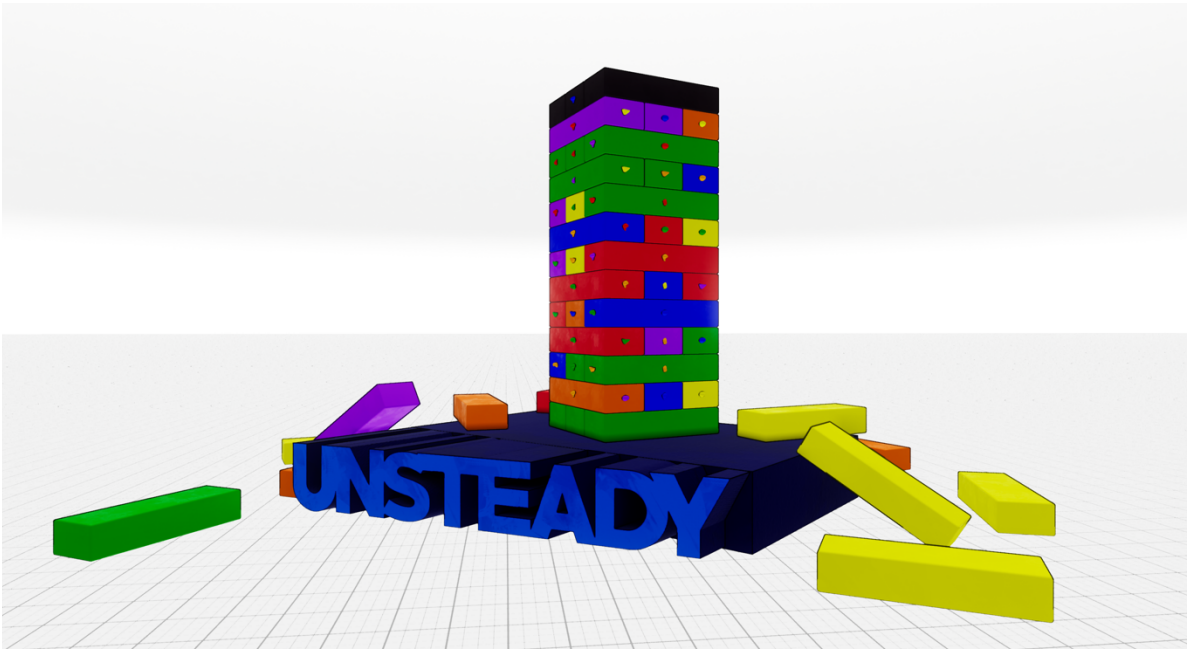
[116] Boucher and Piché, *Sound/image relations in videomusic*, 17.

Chapter 5

Compositional Portfolio – *Unsteady VR*

Unsteady VR (*Figure 28*) was developed by two independent creators, Kyle McKee and Tyler Campbell, who contacted me in April 2020 to compose a soundtrack for their game. *Unsteady VR* is a puzzle focused videogame made for virtual reality which emulates the table-top block game *Jenga*. Players must remove as many blocks from the tower as possible without it collapsing. Their size relative to the tower means they must climb it to reach certain blocks, necessitating careful consideration before actions are made. The game is broken up into ten 'worlds', each with ten levels. Each level has a *Jenga* tower which increases in size and complexity. The player is awarded points for removing blocks from the tower without it collapsing and must earn enough points to be awarded a star. Each level has three stars which can be earned by the player. These stars represent the difficulty tiers of each level, where achieving 1 star is easier than achieving 3. They must earn a certain number of stars before the next world is unlocked.

Figure 28 – A Screenshot Of *Unsteady VR*



I began writing the soundtrack for *Unsteady VR* while the game was in the pre-production phase. While design documents weren't created for the game, game director Kyle McKee provided small briefs on the number of tracks I was expected to compose and the kind of music he wanted for the game. These briefs were one or two sentences in length and included

reference tracks from other games with descriptions of the worlds they were intended to accompany. I provided some initial sketches for the Main Theme and Robot Boss as a starting point while planning the rest of the soundtrack with the developers.

5.1 Soundtrack Plan

The design briefs from the developers conveyed that each game world would match a different aesthetic, and that the musical content should match. Each track was titled based on the world it was designed to fit. Two boss levels were included which would feature their own music:

- Living Room
- Moon World
- Arcade World
- Factory Boss
- Star World
- Sky World
- Water World
- Volcano World
- Robot Boss
- UFO Boss

The budget and resources dedicated to *Unsteady VR* meant that the developers could only afford one track for each world. Each track was to be 2-3 minutes in length and would loop endlessly until the player completed all 10 levels, meaning the player would be exposed to the same musical material for a considerably long time. This meant variation would need careful consideration at this stage of planning. While varying each enough to maintain player interest was challenging from a compositional standpoint, the alternatives were either creatively or financially unfeasible for the developers. The priority was for music to match the visual aesthetic of each world, so creating generic, interchangeable tracks wasn't an option to vary the music. Creating shorter tracks for each level inside each game world was a possibility, but with ten worlds at ten levels each, this would mean a hundred or more individual tracks, which would take longer for me to compose and for the developers to integrate. The developers didn't like the idea of longer ambient tracks, even though they would disguise potential repetition from the player, as they were unlikely to motivate the player in the same way.

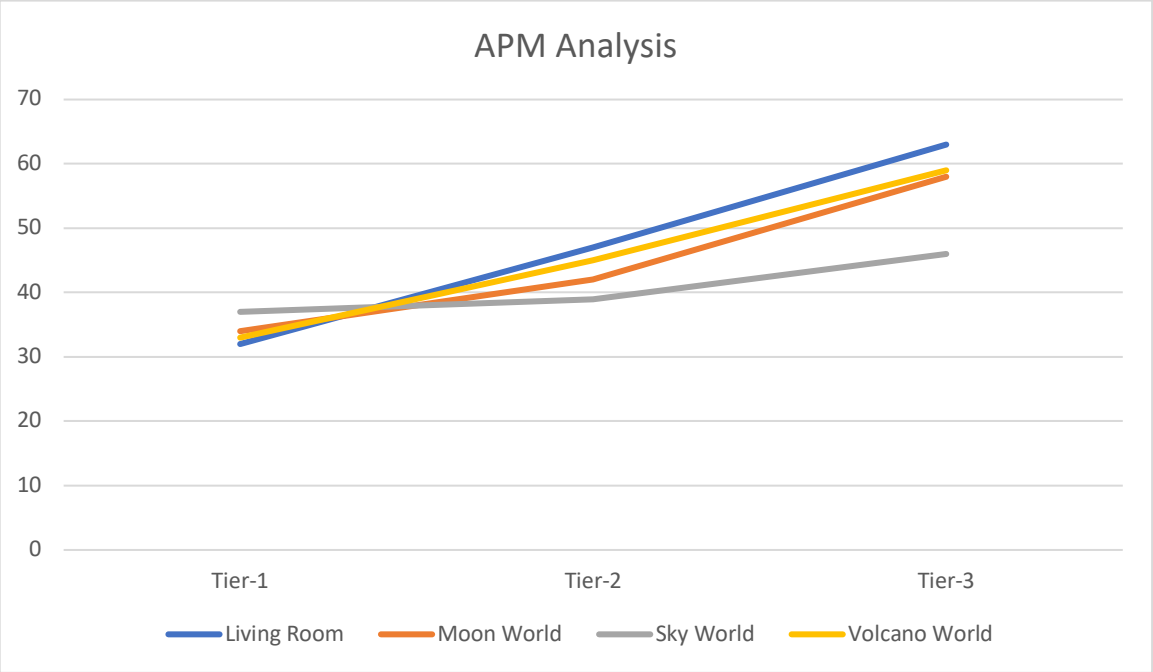
In addition to the structural difficulties related to planning the soundtrack, the strictly puzzle focused nature of *Unsteady VR* made it difficult for me to approach the music from a thematic standpoint. For instance, when composing for previous games, I could scatter a main theme throughout the soundtrack and easily associate it with the lead character. I could reuse the

protagonist's motif and vary dynamics, timbre, and texture based on the context provided by the story. *Unsteady VR* featured no visible protagonist, making it harder for me to ground any motifs to a character's perspective in the same way. Consequently, I decided early on in the process not to use leitmotif as I did with previous games and to create standalone tracks with no continuity in themes. I anticipated that this approach would give me more freedom to link the style and expression of the music to the game worlds and create as much variation as possible so as to not exhaust the player. While this did give me a starting point for composing, I found it difficult to hinge particular creative decisions on the game worlds without divorcing the music from the gameplay. How would the music increase or decrease in amplitude or texture without an overarching context connected to the player? This is where I started using intensity measurements to gather more information on *Unsteady VR* that I could use to guide composition. After the first two tracks were sketched using developer briefs, I was provided with an early gameplay demo of four playable worlds. I began surveying these worlds for their visual behaviour and interactivity to determine how intensity changed during *Unsteady VR* in the absence of story events.

5.1.1 Interactivity Analysis

After measuring the APM of a few worlds I realised that there was more interactivity variation in the difficulty tiers for each level than in between the worlds themselves. For instance, the APM of the Living Room averaged around 32 during the Tier-1 level of difficulty, moving to 47 during Tier-2 and 63 during Tier-3. The APM for the Moon World was 34 during Tier-1, 42 during Tier-2 and 58 during Tier-3. The APM for the Sky World and Volcano world differed only slightly from the other measurements, showing that the primary changes to gameplay intensity were similar for the full duration of the game. This meant that the shifts in intensity between each level's tier of difficulty would be same between each game world. Therefore, a link between the music and these difficulty tiers would be a feasible area to vary tracks in each of the ten levels for each game world. The correlation between interactivity in each tier can be seen in the following graph (*Figure 29*).

Figure 29 - A Graph Showing The APM In Four Of *Unsteady VR's* Worlds



This was the eventual reason for why each level has three musical variations and is the primary way intensity was used as a compositional prompt. The table below contains the notes I collected for each tier of *Unsteady VR's* difficulty.

Table 1 – Unsteady VR Difficulty Notes

Tier 1 (1 – Star)	The one-star variant of the levels is easier than the other tiers. Button inputs are made when grabbing blocks for either climbing or dismantling. Analogue movements are made when moving or reaching out to grab a block. Fewer blocks are needed to be removed before the tier is complete, and therefore fewer inputs are required to complete the tier. Conclusion: Low Input-Requirement
Tier 2 (2 – Star)	The two-star variant of the levels are harder than the one-star tiers. More button and analogue inputs are required as more blocks must be climbed to complete the level. Conclusion: Medium Input-Requirement

Tier 3 (3 – Star)	The three-star tier is the hardest of the three. More inputs are required to move a single block, to prevent interconnected blocks causing the tower to collapse. The higher number of inputs required per-block means this tier has the greatest number of inputs when compared to tiers 1 and 2. Conclusion: High Input-Requirement

5.1.2 Visual Behaviour Analysis

Unsteady VR's recorded gameplay videos were analysed for PCR readings in the same way as the examples presented in Chapter 3. I conducted an on-screen behaviour analysis in the same difficulty-based format as the interactivity analysis. The only environmental changes in each level were centred around the tower immediately in front of the player. Considering the difficulty tiers increase as the player removes more and more blocks, there is less stability to the tower the longer a level progresses requiring the player to look at the tower as a whole more often to determine the most appropriate block to remove without it collapsing. The higher the difficulty, the more unstable the tower becomes, and the more the player looks around to make decisions. The head-mounted camera is the central viewpoint for the player, and therefore displays higher visual behaviour as the player moves their head more frequently. This resulted in the visual intensity analysis reflecting similar information to the interactivity analysis, where the most notable changes occurred as the difficulty increased. All other environmental assets were non-interactive, non-animated and identical between each tier of gameplay difficulty.

Table 2 – Unsteady VR Visual Behaviour Notes

Tier 1 (1 – Star)	The environment immediately in front of the player contains the tower to be climbed and dismantled. Each block in the tower is a different colour. Grab-points the player can use to climb the tower are visible on each block. The environment beyond the tower appears distant and large. The first environment is a giant living room complete with furniture and household items. The tower and the platform it stands upon are the only listed objects which are interactive. All

	distant environmental assets are not interactive in any of the tiers. Conclusion: Low Visual Behaviour
Tier 2 (2 – Star)	The tower is more unstable, but sturdy. More of the surrounding environment is visible due to the gaps in the tower. This means more of the environment is visible as the player moves their head, creating more colour and contrast variation than the 1st-Tier. The player must investigate the tower more thoroughly to determine the next block to grab, resulting in more frequent head/camera movement. Conclusion: Medium Visual Behaviour
Tier 3 (3 – Star)	The tower is considerably unstable with various gaps. Most of the obvious blocks have been removed, meaning appropriate blocks for selection are harder to find. Every attempt to remove a block can potentially collapse the tower, meaning the player must look at many points on the tower to determine the overall impact of their actions. Head/Camera movement has increased from the 2nd-Tier. Conclusion: High Visual Behaviour

5.1.3 Analysing Additional Stimuli (AVUs)

Having only sourced limited information from the game's interactivity and visual behaviour, I sought information on the AVUs present within gameplay. Considering the visual behaviour and interactivity of the 1-star difficulty tier was lower than the 2 and 3-star tiers, it appeared likely that the number of AVUs would reflect a similar dynamic. This dynamic could be identified by evaluating the AVUs present across the game as a whole and then determining whether or not the difficulty tiers changed the number.

Table 3 – AVU Analysis

AVUs:	Instantaneous	Short/Medium	Long
All known elements that have an audiovisual representation in-game.	- Grabbing Blocks - Block Collision - Block Removal - Player Footsteps - Player Collision	(none)	Ambient SFX Loops (world dependent)

Considering the in-development status of *Unsteady VR* during this analysis, sound effects beyond those listed in the instantaneous AVU category were not implemented. The developers informed me that they would source suitable ambient sound-effect loops independently. In the Casino World, for example, the sounds of dice rolls, coinage, and crowd chatter can be heard. In the Factory World, machinery, servo motors, and distant metallic clangs make up the ambience. This prompted me to organise the long AVUs into a table of their own to evaluate the frequency space they would occupy to determine any sounds which may clash with the music I intended to write. The videos included with this portfolio still show *Unsteady VR* in an unfinished state. The full list of sound effects below are still yet to be implemented, which is why some of them are not audible while viewing.

Table 4 – Long AVU Analysis

World	Ambient SFX Loop Features	Frequency Space Occupied
Living Room	- Birds Chirping - Cars Passing - Distant Pedestrians	Between 750 – 2500Hz
Casino	- Slot Machines - Coinage - Crowd Chatter	1000Hz and above
Factory	- Machinery - Servo Motors - Metallic Clanging/Banging	750Hz and above
Robot	- Roaring (From Robot) - Explosions	Between 75 and 2500Hz

	- Stomping (Heavy Robot Footfalls)	
Volcano	- Fire - Hissing/Bubbling (Lava) - Low Rumbling	Some sounds between 20 – 75 Hz and others above 2500Hz
Arcade	- Arcade Machines - Coinage	2500Hz and above
Water	- Gulls - Waves Crashing	Some sounds between 100 – 250 Hz, the rest above 2500Hz
Sky	- Wind blowing - Eagle/Hawk Calling - Plane Flyby	750Hz and above
Moon	- Astronaut Radio Chatter - Low Rumble	Some sound between 20 and 75Hz, others between 2500 and 7500Hz
Star/UFO	- Passing Spaceships - Alien Spaceship - Shooting Stars	Sounds occupy low, medium, and high frequency bands

Having sourced a comprehensive list of AVUs and a consistent intensity structure for each world, I discussed potential implementation methods with the developer to ascertain our next steps.

5.2 Implementation Planning

The results from my interactivity and visual behaviour analysis indicated that the difficulty tiers were the most practical means of varying the music. The AVU analysis revealed the kind of music I could put to the game without overloading the player and clashing with the intended sound effects. I used the information I had gathered to suggest to the developers that the music vary by dynamics and expressiveness as the player progressed from one tier to another. This decision was made considerably late into my work on the soundtrack, after I had completed

five individual world tracks. After this point, each track would have three variations, one for each tier, but would maintain their structure, harmonic and melodic content so seamless transitions could be created between each. The developers concurred with the suggestion, leading us to plan three variants for each composition suited for each tier of difficulty.

The following sections explain the compositional process behind each track in the *Unsteady VR* soundtrack. Each section will contain a description of the in-game environment, followed by an overview of the developer's instructions given prior to composing the first sketches. I will then describe how the initial track was composed before explaining how the initial track was split into intensity variations.

5.3 Living Room/Main Theme (Gameplay Showcase | Composition Showcase)

Figure 30 – A Screenshot Of The Living Room World In *Unsteady VR*

The "Living Room" (*Figure 30*) is the first playable world in the game, set in a typical lounge room in a suburban house. The player starts at regular size and shrinks down upon starting the puzzle gameplay.

5.3.1 Living Room/Main Theme Brief

The developers decided that that *Unsteady VR*'s Main Theme would play over the top of the Living Room, so it was the initial music the player heard upon starting. To account for this dual purpose, I planned to involve overarching concepts related to *Unsteady VR* as a whole game in addition to aesthetics communicated specifically by the Living Room. Despite the low energy of the domestic environment, I deliberately attempted to heighten the excitement of the track to emphasise the fun posed by the simple, carefree nature of the challenges yet to be experienced by the player. This was intended to reinforce engagement with the puzzles and encourage the player as they progressed.

5.3.2 Living Room/Main Theme Composition Process

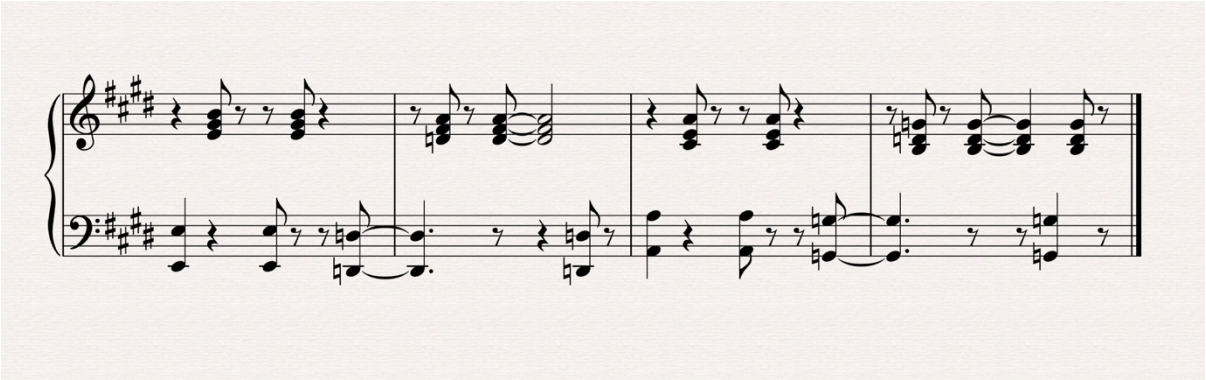
After setting up the Logic X project with a suitable tempo and time signature (140bpm, 4/4), I created a looping kick drum that played for the full duration of the track. A limiter patch placed on the stereo output helped provide a consistent rhythmic device to unify the piece. This is a technique I developed while writing action music for previous games and features in 8 of the 10 tracks written for *Unsteady VR*. The ducking effect comes mainly from the EQ placed over the kick: a low-pass filter with the gain turned up to drown out any neighbouring instruments (*Figure 31*).

Figure 31 - Kick Drum Limiter/EQ Configuration

After setting up the rhythmic foundation, I sketched a looping bassline designed to run alongside the kick throughout the whole piece over which additional content would be layered. Each layer iterated above these looping elements varied in instrumentation and melodic content, communicating the structure of the piece through variation while the bass and percussion remained static. I developed the bassline into a set of chords (*Figure 32*) to

accompany the melodic devices I intended to bring to the piece, finalising the core structural elements to the piece before effectively improvising the remaining content.

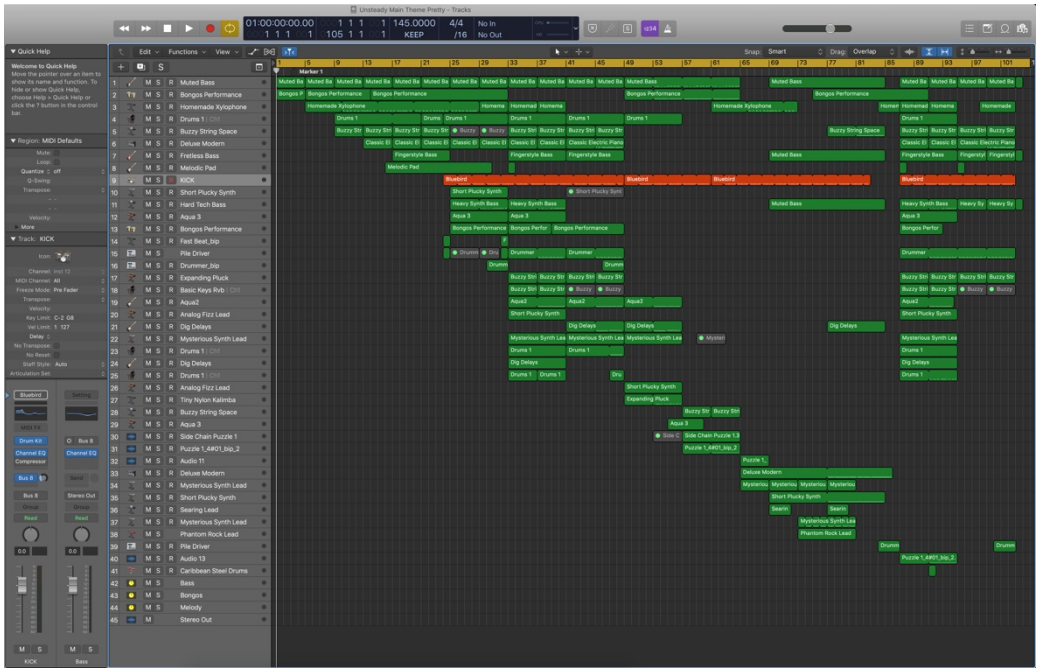
Figure 32 – Chords And Bassline For The Main Theme



Instruments were selected to match the Living Room setting: Brightly lit, open windows, furniture and scattered household items. Light acoustic percussion reinforced the 'woody', 'earthy' tone-colour I wanted to bring to the world to heighten the domestic imagery. Even when introducing synthesizers or adding digital delay, reverb or filters to the acoustic instruments, the timbre of the instrumentation adhered to the relationship between the visuals and the atmosphere of the Living Room environment.

All variants of the Main Theme are derived from the high-intensity variant which was composed first. Despite this link, each variant differs in structure and length. The high-intensity variant is segmented into four sections: an introduction, a statement of the main motif, an interlude, a restatement of the main theme, a second interlude, and then a concluding repetition of the main motifs. The relationship between bass and percussion is the driving energy behind the composition, staying consistent as the piece grows layer by layer. The kick emphasises first interlude and continues at full volume until stopping for two bars before the final repetition to re-invigorate the energy of the piece in the final bars (the red highlighted regions in *Figure 33*).

Figure 33 – The Final Structure Of The Main Theme With Highlighted Kick

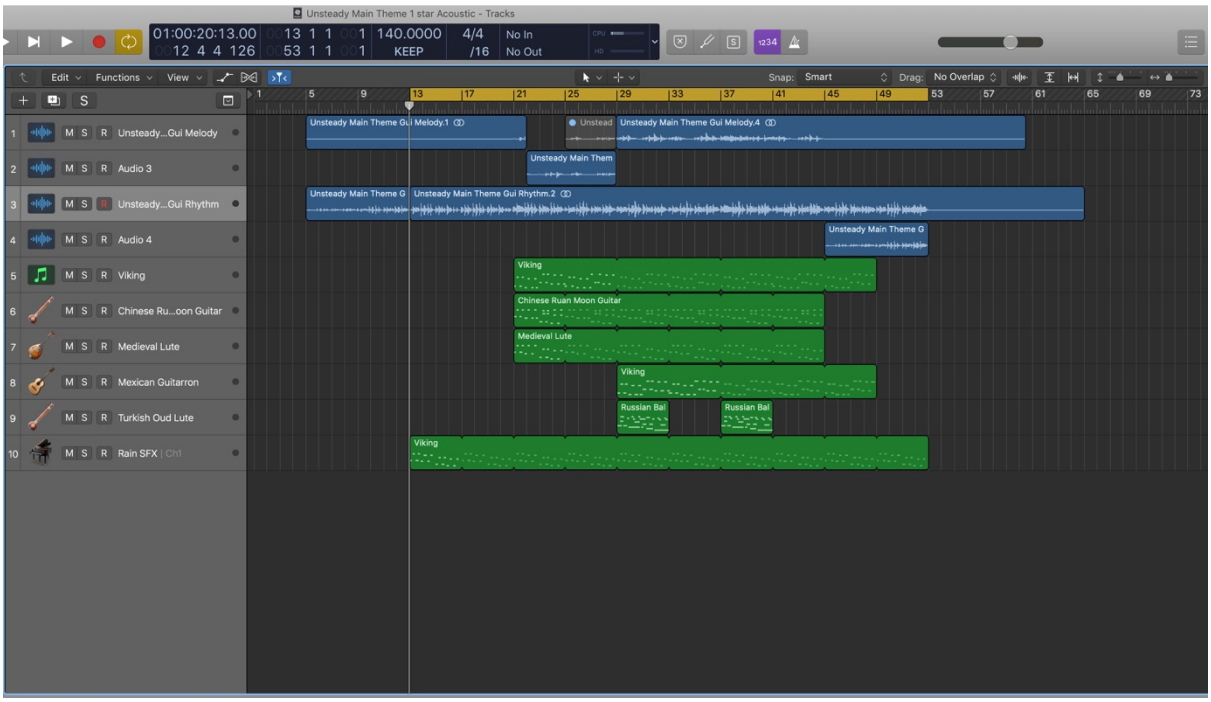


The second interlude serves as a break from the rest of the piece, featuring basic pop chords and non-specific melodic elements as 'filler' before ramping up for the final section. The piece contains its own dynamic and structural variation to suit multiple contexts. As a main theme, the track will likely be played over marketing material, in-game menus, or credits.

5.3.3 Living Room/Main Theme Intensity Process

The decision to vary this track was a combination of two choices, the second made later in development. The first was an idea from the developers to create an 'unplugged' version of the main theme specifically for a trailer. The second was the introduction of the three-tier difficulty system which reintegrated the unplugged version into the game as the "Main Theme Low-Intensity" track (*Figure 34*). As a result, the process of varying the Main Theme started with the high-intensity version before moving onto the low-Intensity version. The medium-intensity version was created last, and integrated elements from the High and Low-Intensity versions to bridge the gaps. The resulting work created a benchmark for the contrast between low and High-Intensity, where the latter would feature a thinner texture with fewer instruments and lower amplitude overall. This was the first track to be varied in this way and influenced my approach to varying the other tracks in the soundtrack.

Figure 34 – A Screenshot Of The Low-Intensity Main Theme Variant In Logic Pro X



To create the 'unplugged' sound recommended by the developer, I re-recorded the chords and melody form the High-Intensity variant on acoustic guitar and placed them in their own project. Due to my inexperience with acoustic guitar, I had to record at a slower tempo and then edit the recordings to match the final tempo. This required careful work to preserve the natural sound of the notes, as time-stretching creates additional noises or edits that change the original sound. This was particularly important for any use of vibrato, where the movement between notes becomes sped up, sounding un-natural. Any incorrectly plucked notes required reinforcement with additional digital elements to mask mistakes and maintain the atmosphere created by the High-Intensity version. To achieve this, I used a number of Logic Pro's included ethnic virtual instruments due to their similar sound. This allowed me to place reverb or delay on the more consistent, stronger sounding notes provided by the virtual instruments to de-emphasise any errors in the recorded guitar. These extra steps were taken to ensure that the guitar samples sounded natural and realistic, an aesthetic I felt necessary to communicate the tone requested by the developer.

The Medium-Intensity version was composed last after the decision to change the music via the difficulty system. The main differences between it and the other two variants were primarily structure related. Only the acoustic guitar version of the primary melodic motif was taken from

the Low-Intensity version, and all other elements were sourced from the High-Intensity version. The texture is notably thinner than the High-Intensity variant, with several percussion and synthesizer elements removed.

5.4 Sky World (Gameplay Showcase | Composition Showcase)

The Sky World (*Figure 35*) has the player deconstructing towers that float above the clouds. The environment is notably bare compared to the Living Room, with a cloud-laden skybox being the only visible content other than the menu elements and puzzle towers. At random points throughout the gameplay, a giant plane flies towards the player, exploding if a tower is present. This poses no danger to the player and does not change the gameplay in any way. For these reasons in addition to instructions from the developer the plane is not acknowledged by the soundtrack.

Figure 35 – A Screenshot Of The Sky World In *Unsteady VR*

5.4.1 Sky World Brief

The brief for the sky world was simple, with the one reference track being the primary source of mood and instrumentation. While the general aesthetic of the Sky World and the reference track provided was conceptually easy to picture, I found this track the hardest to compose. The low energy was the single contrasting element that separated the Sky World track from the previous ones created until this point. Every track prior to the Sky World piece could be unified

under percussion which followed faster tempo (100bpm and above). While the final tempo for this track in the Logic X project file was 120bpm, the downbeats and percussion elements throughout the piece makes it feel as if it is set to 60bpm. The general mood for this piece was to be reflective and poignant to support the emotional concept behind the reference track, a melancholy song dedicated to a lost loved one. The developer stipulated this track was to be happier sounding.

5.4.2 Sky World Composition Process

For almost every other piece on the *Unsteady VR* soundtrack, I had the advantage of being able use several short synthesizers overlayed over the top of a looping drum track. Delay and reverb effects extended any staccato notes played by synthesizers to thicken the texture of the piece and I could simply improvise melodic content or restructure the piece until it sounded complete. The Sky track felt like the complete opposite, with a slow pulse and little room to involve repeating synth ostinatos. I plotted the initial sketch to be as simple as possible. I sketched a I, V bassline as a starting point so I could improvise melodic content over the top. The main method of building the track's texture involved several layered pad synthesizers and plucked ostinatos played on harps, guitars, and ethnic stringed-instruments. These were sourced from Logic X's suite of acoustic virtual instruments, which could be extended using delay and reverb to fill out the texture. This built enough harmonic foundation for me to plot a melodic motif that served as the primary melody for the track.

5.4.3 Sky World Intensity Process

When it came to varying the intensity for each track, we labelled each of the completed tracks as either low, medium or high-intensity and then determined how the other variants would change. The completed Sky World track was determined to be the low-intensity variant. To vary this, the developer recommended adding harps and banjos to intensify the existing acoustic instruments which played plucked melodies. Considering my difficulties with the slower tempo and elongated melodies, my process of intensifying the sky track involved filling any gaps of silence with simple rhythms or ostinatos to thicken the texture and increase the dynamic range. In the medium-intensity variant, I used harps and additional stringed

instruments to provide mid-register, semi-quavered alto melodies throughout the full duration of the track. For the high-intensity variant, I reinforced these melodies with synthesizers emulating plucked stringed instruments playing the same material. Additionally, a plucked synth bassline was incorporated into the high-intensity variant to give further energy to a previously neglected lower register. The final statement of the main motif in the latter third of the piece was bolstered using banjo notes, playing alto accompaniment to the soprano already present in the low-intensity variant.

5.5 Moon World (**Gameplay Showcase** | **Composition Showcase**)

The Moon World (*Figure 36*) uses visual clichés from Apollo 11's 1969 Moon landing, situating the player on a rocky lunar surface with the Earth behind them. The world gravity is halved, meaning physics interactions between the player and the game world feel slowed, emulating moon-gravity.

Figure 36 - A Screenshot Of The Moon World In *Unsteady VR*

5.5.1 Moon Brief

The Moon World piece was the final track composed before the variation process began. At this point in development, the *Unsteady VR* soundtrack had one track for each world with no

intensity variation. The Moon track was described by the developers as needing to be 'low-fi but energetic enough to fit with the other, more up-beat worlds'. This in addition to the reference tracks gave me an instrumental starting point and allowed me to tackle the Moon world piece in a similar way to the factory, volcano and living room tracks.

5.5.2 Moon World Composition Process

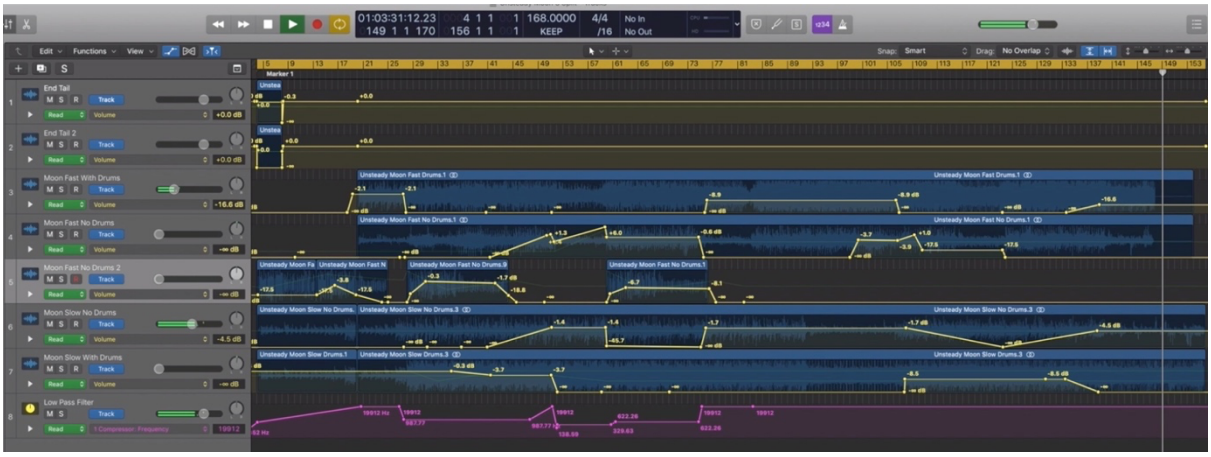
I created two sketches for the Moon World before the final track was approved. The first of these sketches was considerably more upbeat. Bright diatonic melodies played over the top of a high-activity electronic drumline, creating an overall heightened energy which did not suit the slower atmosphere created by the Moon imagery. The developer concurred that this version wasn't entirely appropriate for the environment, but enjoyed the instrumentation, recommending I continue down that path. I began a second sketch by deleting most of the content and creating new material using the same instruments. The resulting sketch became the high-intensity variant of the final Moon World track. This version was created by layering melodic elements over the top of a looping bass and drum line, with melodic interest communicated through a mix of descending keyboard scales played on bright synthesizers to evoke starfield imagery while the rhythm and bass drove the energy of the piece. These runs were enhanced using delay to create interest through rhythmically precise harmonic clashes. A low-pass filter is automated to fade in and out, muffling the entire track to create a sense of 'pressurisation'. I wanted to give the illusion that the player was listening to the music from within a space suit which could potentially impact sounds.

5.5.3 Moon World Intensity Process

The process of modifying the Moon World music into create the low and medium-intensity variants was unique compared to the rest of the soundtrack. The track which resulted from the second sketch was determined to be the high-intensity variant out of the eventual three tracks. To create the low-intensity variant, the developer specifically asked me to simply slow the tempo of the track by half. I reduced the tempo of the original track (168 BPM) by half and modified the kick to appear on the 1st and 3rd beats of each bar. These were very minor changes which resulted in a considerably different sound. The medium-intensity variant was a simple

combination of the high and low-intensity variants, overlayed on top of each other with volume transitions between material from both tracks. The low-pass filter aided in masking these transitions, conveniently masking any clashes that arose from merging two tracks with conflicting tempo settings. I kept the exact automation structure from the high-intensity variant, unifying the dynamic movement between the two (*Figure 37*).

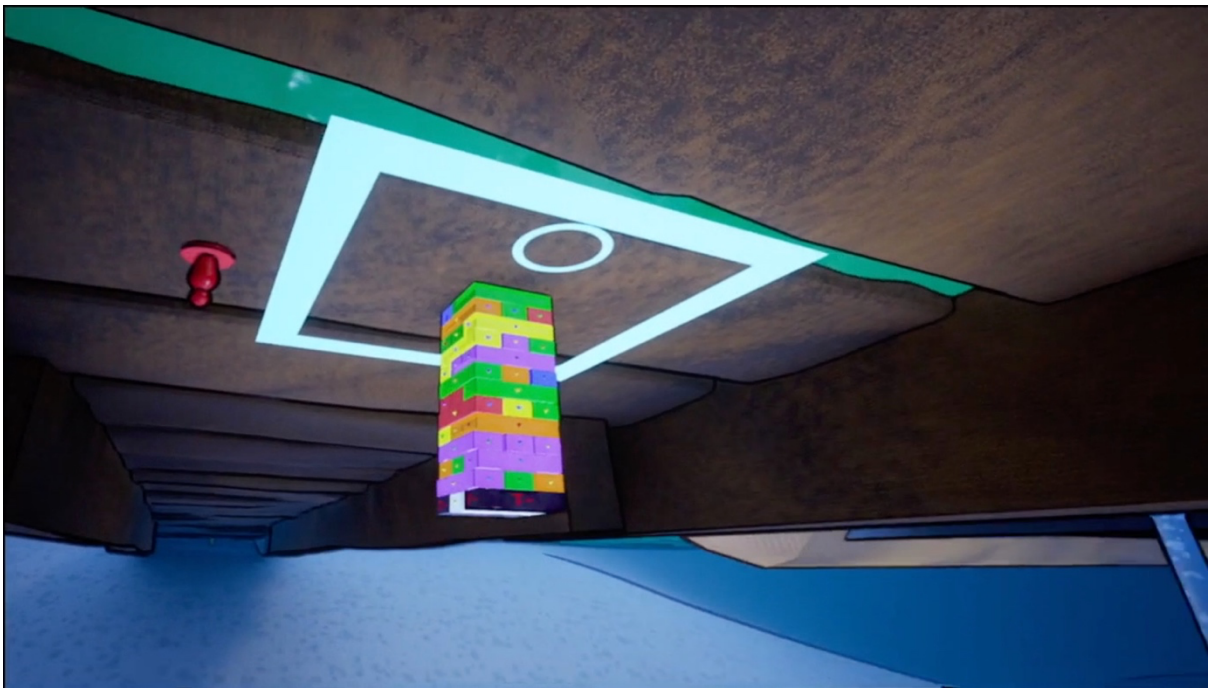
Figure 37 – A Screenshot Of The Medium-Intensity Moon World Variant In Logic Pro X



5.6 Water World (Gameplay Showcase | Composition Showcase)

The Water World (*Figure 38*) has the player dismantling a tower underwater beneath a dock. Any blocks dropped by the player float to the top of the world, a result of flipped gravity. This variation creates an interesting visual aesthetic while bringing a unique complication to the puzzle gameplay.

Figure 38 – A Screenshot Of The Water World In *Unsteady VR*



5.6.1 Water World Brief

The first sketch for the Water World was actually sent to the developer as an attempt at the Sky World track. The developer indicated they liked the track, but that it would be more suited to the aesthetic they'd planned for the Water World. This essentially bypassed the briefing process, as the sketch was robust enough to be approved without any additional direction.

5.6.2 Water World Composition Process

I built this composition over the top of the plucked baseline - Embertone's pizzicato patch for the *Blakus Cello* virtual instrument. *Blakus* samples are considerably noisier than my other Cello libraries, featuring less reverb or EQ enhancements. I combined the buzzy pizzicato bassline with some light djembe percussion to create a looping bassline of short, easily identified notes to improvise content over the top. This bassline is consistent throughout the low and medium-intensity variations of the piece. I wrote most of the melodic elements on a piano enhanced with elongated reverb, filling out the high register to contrast with the bass elements. The reverb helped fill any gaps of silence with high-end reflections, thickening the

texture while keeping the total instrument count low. I book-ended the piece with a looping celesta ostinato which creates a sort of hazy, dream-like soundscape when paired with the piano. This mix of emphatic, staccato bass notes with elongated, soft melodies created the two categories for additional instruments to fit under as the composition grew.

5.6.3 Water World Intensity Process

The developer described the approved version of the Water World track as the high-intensity variant and suggested incorporating a similar low-pass filter technique from the Moon World piece to create the feeling of being underwater. The instrumentation used in the Water World piece up until this point did not sound as effective with the low-pass filter, but the idea held value. Instead of altering the sound of the existing instruments, I decided to replace them to create the alternate variations. I began by creating the low-intensity variant, replacing the plucked bassline from the high-intensity variant with shorter bass notes at the beginning of each bar. This bassline was composed using a Symphobia virtual instrument from Project Sam, providing a combination of double basses, trombones and tubas. These plucked notes were far heavier in bass frequencies and created anticipation when paired with the reduced rhythm. I complimented the new bassline with piano chords and entirely removed the kick from the piece to round out the full low-intensity variant. The medium-intensity variant was a matter of re-integrating the instruments I had removed from the high-intensity variant to bridge the gap between. This included the kick, the bouncing *blakus* bassline and some extra cymbals.

5.7 Arcade World (**Gameplay Showcase** | **Composition Showcase**)

The Arcade World (*Figure 39*) is inspired by many videogame clichés present in classic 1980's video arcades. The environment is a brightly coloured room with several arcade machines scattered throughout. The platform upon which towers are placed sits inside an arcade cabinet in front of the player, creating the feeling that the levels are taking place 'inside' a classic videogame.

Figure 39 – A Screenshot Of The Arcade World In *Unsteady VR*

5.7.1 Arcade World Composition Process

No brief was provided for the Arcade World composition. I was instead instructed to 'surprise' the developer, and that the levels would 'take place inside a PONG machine, if that helps.' I built the piece by searching for instruments which emulated 8-Bit soundtracks from early videogames. I found a number of synthesizers in Logic X's included virtual instrument library which matched the aesthetic I was looking for, allowing me to sketch a looping bassline similar to the other tracks. The three elements which underpin the Arcade World composition are the bassline, two main melodies and some accompanying looping ostinatos. The bassline remains the same throughout the piece, accompanying the melodies which structure the piece. The ostinatos help create variation in gaps where the melody isn't present while contributing extra 8-bit sounds to the overall piece.

5.7.2 Arcade World Intensity Process

The approved version of the Arcade World track was allocated as the high-intensity variant. The developer instructed me to create the low-intensity variant of the track by lowering the fidelity of the instruments, so they sounded more 8-bit. Almost every instrument in the high-intensity variant had a degree of reverb or delay added, meaning that the process of reducing the track for the low-intensity variant mainly involved turning off these effects. The resulting low-intensity variant is considerably drier, with a few ostinatos removed to thin out the overall texture. Additionally, I placed a 'bitcrusher' distortion effect over the top of the master fader in Logic to give the mix a crunchier sound. The process of creating the medium-intensity variant was more simple, and involved reintroducing the elements I had removed from the low-intensity variant. This included instruments and effects. I opted not to incorporate the bitcrusher distortion, and instead separate the medium-intensity variant from the high-intensity variant through differences in the percussion. The medium-intensity variant possesses only the kick drum, with the high-intensity variant featuring a full range of cymbals, snares, and toms.

5.8 Casino World (**Gameplay Showcase** | **Composition Showcase**)

This environmental setting would be a Casino interior complete with poker machines, patrons, and roulette tables (*Figure 40*). The visual aesthetic of the world was to be bright, busy, and grandiose, evoking imagery of 1960's crime and spy movies.

Figure 40 – A Screenshot Of The Casino World From *Unsteady VR*

5.8.1 Casino World Brief

The developer provided very few descriptions of the world itself and instead focused on reference material and a recommendation to incorporate jazz. As a result, I had far more creative freedom with the Casino track compared to the other tracks, giving me the opportunity to focus more heavily on harmony and instrumentation to maximise the impact of what would be a departure from the genres depicted by the soundtrack thus far.

5.8.2 Casino World Composition Process

I aimed to emulate 60s big band music using a series of virtual orchestral instruments behind a series of piano melodies. This choice of genre allowed me to build the track using more

traditional compositional approaches, structuring the motifs in a song-like format. This was similar to the approach I used with the Main Theme/Living Room track, where the piece was composed as a standalone work in addition to serving as gameplay accompaniment. As with the previous tracks, I composed a looping set of bass and rhythmic devices (pizzicato double bass with some light cymbals) and then created some piano improvisations which eventually formed the main melodic motif. I composed three additional sections and then alternated between them to create a varied piece. The primary orchestral instruments for this track were sourced from the Project Sam Symphobia library for Kontakt.

5.8.3 Casino World Intensity Process

The variation process for this track took place under a tight deadline, with very few steps taken to achieve the final result. The developer determined that the completed version of the track was suitable as the high-intensity variant, tasking me with creating the low and medium-intensity variants. To achieve this, I simply selected every region in the Logic Project and reduced the velocity of each midi note by about 15%. I chose to remove instruments from the percussion section so as to not compromise the texture of the chords I had constructed using the orchestral instruments. I completely removed them in the low-intensity variant, and removed the toms, kick and snare from the medium-intensity variant. This brought more focus to the piano, particularly in the low-intensity variant. I increased the volume of the piano parts in the piece and removed some orchestral parts from both the low and medium-intensity variants to reflect this change of focus.

5.9 Factory World and Volcano World

Factory (Gameplay Showcase | Composition Showcase)

Volcano (Gameplay Showcase | Composition Showcase)

The Factory World (*Figure 41*) has the player dismantle towers on a moving conveyer belt, with hostile machinery, lasers, and fire passing by. The developer brief highlighted a need for frantic, high-energy music which communicated the energy of the world. While a sketching process identical to the Main Theme and Sky World compositions resulted in an initial track approved by the developer, this track was rearranged to fit the Volcano World, which then created unexpected complications later down the line.

Figure 41 - A Screenshot Of The Factory World In *Unsteady VR*

5.9.1 Factory World First Attempt

The developer brief described the music as needing to be ‘a little stressful, but still have a heavy bass beat to it.’ I created a sketch involving a heavy kick over the top of an aggressive synth bassline. The synthesizer patches were sourced from Logic X’s virtual instrument library, distorted to emphasise the bass while highlighting some grinding specular elements on the high end. The sketch was enveloped using a low pass filter fading in as the piece started and fading out as it ended. The sketch was a mostly successful attempt at communicating energy and mood but lacked the sense of panic the developer had requested. In my opinion, this came down to the selection of the bass notes and lack of melodic motifs. The bass for the sketch followed a ‘i, iii, VII, i’ pattern with melodic elements revolving around the tonic bouncing between octaves. This mostly diatonic chord/melody combination sounded empowering rather than unstable. The developer responded by praising the aggression of the piece but noted that it did not feel “panicky” enough. They said that this sketch would work particularly well as a track for the Volcano World (*Figure 42*) and asked if I could make a few changes to make it suit before beginning work on a second attempt at the Factory World music. I concurred and began work on the second sketch.

5.9.2 Factory World Second Attempt

I opted to start a new version of the Factory world track from scratch, implementing a kick/limiter combination from the other compositions and then by creating an electric bass bassline that mainly stayed to the tonic with the occasional inclusion of the minor 2nd scale degree. This introduction of chromaticism immediately brought about a sense of instability, thus creating a basis for the panic intended by the developer. I extended the chromaticism to be included in the melodic motifs, this time communicated by extensive electric guitar work. By incorporating the thematic intention of the developer as a harmonic device, I was more successfully able to meet their expectations throughout the piece. This contrasted with the first sketch where I had attempted to communicate the sense of panic through instrumentation and texture, opting to leave the harmony until last. The developer immediately latched onto this composition and praised it as the best work they had received. I made a few more edits to the track (increasing the tempo and adjusting the timing of some of the guitar recordings) and moved onto the next track.

Figure 42 – A Screenshot Of The Volcano World In *Unsteady VR*

5.9.3 Naming Convention Complication

At this point in development, I had created four approved tracks for *Unsteady VR* and had named the Logic X project files using the name "*Unsteady*" followed by numbers. The first sketch for the Factory World was labelled "*Unsteady* 4.1", and the next was "*Unsteady* 4.2" etc. Despite the number of sketches and tracks I was managing, I only started including the world name in the labels ("*Unsteady* Factory 4.3") after the fifth track. At this point, I began re-naming the older project files to fit this new convention. I reread the developer note to reappropriate the first attempt at the Factory track as accompaniment for the Volcano world but labelled the second attempt as the Volcano World track instead. The resulting fallout from this mistake did not occur until later in development when I began splitting each of the tracks into three variations. When that process began, I added additional elements to the variations of each track to reinforce their aesthetic similarities to the worlds they were written for. The developer only picked up this error after I had spent many hours placing additional 'volcanic' sounding elements into what was actually the Factory World track. After some deliberating, and some failed attempts to realign the two tracks to their intended worlds, we decided to integrate the mistake and move forward. After this point, the first attempt described in 5.9.1 became the Factory World track and the second attempt described in 5.9.2 became the Volcano World track. While the process of planning these compositions was messy, both tracks were approved as the high-intensity variants, allowing me to tackle them in a similar format.

5.9.3 Factory World Intensity Process

In order to vary the Factory World track, I tried to reintegrate a sense of frantic energy requested by the original brief. This was achieved primarily through a mix of electronic and acoustic based percussion. I alternated between cymbal work written using Logic Pro's internal drum libraries and the Native Instruments Epic Dhol percussion library, increasing the percussion with each variant. The low-intensity variant contains a kick, with quieter synthesizers playing the bass. The medium-intensity variant contains more aggressive synth basslines and acoustic percussion from the Dhol library. In both variants, the synth bassline and melodies ducked by the kick/limiter combination from the other compositions create a thinner overall texture on the first and third beat of every bar, allowing me to insert drumming with high reverb to fill this silence and create the sense of a larger space communicated by the tall ceilings in-game.

The high-intensity variant contains the full suite of drums, working with the aggressive synth bass to create the energy requested by the developer.

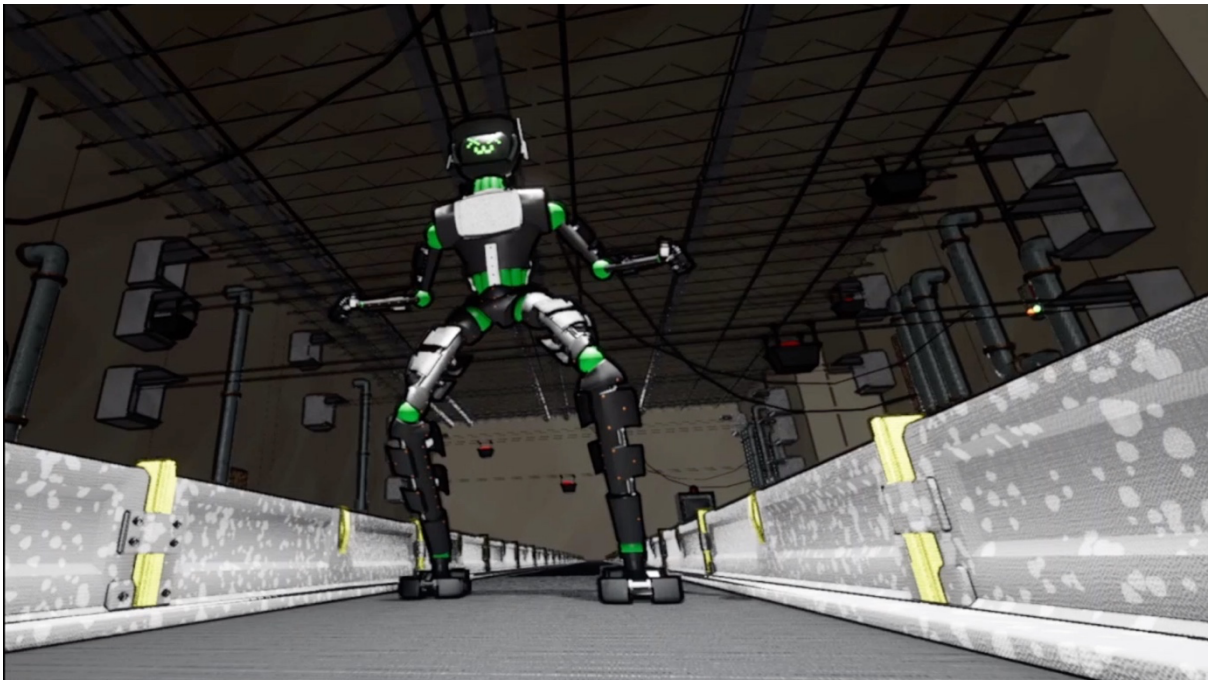
5.9.4 Volcano World Intensity Process

The Volcano World has the player dismantling towers inside an active volcano. The towers slowly sink into the lava below the player, requiring them to dismantle them quickly to avoid sinking with them. The frantic atmosphere of the high-intensity Volcano World track was conveyed using electric guitar and synthesizers, allowing me to create variation by introducing additional instruments. The overall approach to this track would have the low-intensity variant sounding the most acoustic, with the aesthetic of the piece becoming more artificial as the player progressed. I favoured anything that would provide a tribal feeling to the soundscape, searching for percussion or vocal elements to add extra flavour. I used the VOXOS library for Kontakt to source non-pitched vocal shouts, adding these to compliment the percussion. In the low-intensity variant, the shouts act as a snare drum, appearing on the second and fourth beat of the bar. These are reintegrated in the latter third of the high-intensity variant. When combined with ethnic percussion sourced from the Native Instruments Dhol virtual instrument, these extra elements create a 'tiki' vibe that contrasts heavily with the electric guitar/synthesizer combination. The medium-intensity variant contains a balance of instruments from the low and high-intensity variants, with the overall velocity of the midi notes reduced slightly to create contrast in dynamics.

5.10 Robot Boss (**Gameplay Showcase** | **Composition Showcase**)

The Factory World is one of two worlds in the game to contain a boss level with its own music. This boss is a giant robot (*Figure 43*) many times taller than the player, stomping back and forth along the conveyer belt as the player attempts to climb it. The player will eventually be able to destroy the robot by climbing to its head and removing a 'battery'. The robot's movement and handholds for climbing are working and visible in-game, but the 'battery' and demise of the robot have not been implemented. The accompanying Gameplay Showcase does not feature the track I composed for the Robot Boss. Therefore, I have artificially overlayed the music to demonstrate what the level may eventually sound like.

Figure 43 – A Screenshot Of The Robot Boss In *Unsteady VR*



5.10.1 Robot Boss Brief

The Robot Boss music was the second track I was asked to compose after having completed the Living Room/Main theme. I mistakenly believed that this was a world in its own right, especially as I was not provided with any demos, screenshots, or video footage at this stage. I would not conduct my interactivity and on-screen behaviour analysis until after this composition was complete, as that was the first point where I received a gameplay demo. For these reasons, I did not know that the robot would be encountered in the Factory World and was not instructed by the developers to create any musical correlation between their soundtracks. The developer brief on this track described as a bonus world requiring high intensity music. The developer specifically requested 'metallic' percussion as an instrumental starting point, carrying forward the frantic atmosphere of the Factory World.

5.10.2 Robot Boss Composition Process

To achieve the metallic sound encouraged by the developer, I immediately started sketching percussion sections using Heavyocity's DAMAGE percussion library for Native Instruments. This library contains recordings of non-instrumental metal objects struck with mallets or sticks.

I created a sketch with a focus on percussion mixed with some additional synthesizer ostinatos, providing ambient melodic backing to the mix of heavy and light drumming. To vary the percussion sections, I would alternate between heavy emphatic drumbeats and shorter, lighter cymbal-like clicks, relying on syncopation and repetition to create variation. While the sketch matched developer expectations, they mentioned a need for higher intensity, ensuring a 'hectic' atmosphere for the player. In response to this, I increased the tempo from 110bpm to 130bpm. This brought a sense of urgency to what had originally been a fairly slower paced track. The lack of melodic content creating an overall thinner texture to the piece made this tempo increase particularly effective. This track was approved as the high-intensity variant of the Robot Boss track.

5.10.3 Robot Boss Intensity Process

The process of creating variants for this piece was a matter of stripping any non-percussion instruments from the low-intensity variant and lowering the dynamics of all instruments in the medium-intensity variant. The high-quality samples provided by the DAMAGE library allowed me to focus on the clarity of the sounds and reduce the piece rather than add or change elements. The medium-intensity variant is almost identical to the high-intensity variant, but with the kick/limiter combination removed and the overall velocity of midi notes for each instrument reduced.

5.11 Star World and UFO Boss

(SW - **Composition Showcase | Gameplay Showcase**)

(UFO - **Composition Showcase | Gameplay Showcase**)

The Star World (*Figure 44*) is the last playable world in-game and was to be split into two parts, a typical world like the others with nine playable levels, and a boss battle similar to the robot. The accompanying track, therefore, is two variations of the same piece: a low-intensity variant for the Star World and a high-intensity variant for the UFO boss. There is no medium-intensity variant. The developer has not provided reasoning for why this variation structure was selected. Furthermore, the Star World is not completely playable, with several bugs preventing the player from scoring points. Additionally, the UFO boss has not been implemented at all.

To compensate for this, the accompanying Gameplay Showcase will demonstrate both variants, artificially placed over footage of the Star World.

5.11.1 Star World and UFO Boss Brief

The initial developer brief requested two entirely different tracks with no crossover in themes. The Star World was described a barebones Moon-like environment stylised to look like a child's drawing, complete with crude stars drawn on a night sky. I was asked to compose 'whimsical' music which conveyed the imagination of a child while setting up some darker tension to foreshadow the eventual appearance of the UFO. The developer gave a separate brief on the UFO Boss, describing a need to correlate it with the Robot Boss through instrumentation. The general idea was to merge themes from the Star World with the percussion of the Robot Boss.

Figure 44 – A Screenshot Of The Star World In *Unsteady VR*

5.11.2 Star World Composition Process

I composed the Star World track first to determine the musical content I would then vary for the UFO track. Despite my difficulty approaching the Sky track, I used similar methods to emphasise elongated melodies to contrast with the Moon track which had similar

environmental imagery. To achieve this, I structured the harmony of the piece to bounce between I and VI, expanding the texture layer by layer. I tried to keep the instrumentation as orchestral as possible, with electronic instruments punctuating particular melodies. This, at least conceptually, was designed to evoke the imagery of a human (orchestral) environment 'invaded' by extra-terrestrial (electronic) beings. I expanded this concept by incorporating a second section with a considerably darker chord arrangement. This helped build the dynamic range of both the Star World and UFO pieces as they approach their latter third.

5.11.3 UFO Boss Composition Process

I started work on the UFO Boss track by heightening the intensity of the Star World track. This involved the introduction of electric guitar riffs and synthesizers which played the same content as the orchestral instruments in the original piece with an added minor modality. This was intended to be preparation for introducing Heavyocity's DAMAGE library I used in the Robot Boss music to correlate them. By the time I started integrating DAMAGE, I found that the rhythmic structure and overall texture to the piece did not blend well with the metallic percussion. I mentioned this to the developer, suggesting I continue treating the piece as a high-intensity variant of the Star World track instead of correlating it with the Robot Boss. The developer concurred, and I continued filling out the track with additional instrumentation.

5.12 Concluding Remarks

Intensity is a structural motivator for several aspects to *Unsteady VR*, pacing challenge and in-game stimuli to maximise the attention and engagement of the player. I was able to resolve difficulties I encountered in composing the game's soundtrack by analysing *Unsteady VR*'s interactivity and on-screen behaviour to determine a larger intensity arc and then use it as a guide to vary the game's music. This overview has showcased the practical application of intensity measurements as a tool to aid the compositional process. It has also demonstrated the way intensity arcs can be used to more closely link music to gameplay.

Conclusion

This thesis has investigated the intensity structures that scaffold videogame design. The unfolding of gameplay events tend to adhere to gradual rises in intensity across the duration of both game levels and a full game narrative. While this investigation suggests that intensity is the result of various interlinked artistic expressions, it further uncovered that intensity can be utilised to inform creative expression as well. Chapters 1 and 2 of the thesis pinpointed the role of intensity as a practical tool during the early stages of videogame development. As evidenced by Collins, the notion of rising intensity is already used in videogame production as a planning and implementation aid for composers and sound designers creating audio assets. By evaluating Collins' work alongside Lopez and Peerdeman, this thesis has mapped out how intensity looms over creative work in the videogame industry undertaken by established industry professionals. This is further evidenced in Chapter 3, where the kinds of AVUs found in released games are shown to constellate in line with intensity arcs. In the pursuit of further integrating intensity as a production tool, I developed two novel measurement techniques to analyse gameplay for changes in intensity. These measurements were integrated into the Gameplay Intensity Framework, which encourages artists to utilise existing intensity structures as guides for tailoring music to the virtual experience. This framework can be used by composers to source compositional prompts from videogame footage or demos which will align their music with the intensity of the analysed gameplay. The Gameplay Intensity Framework was utilised to compose and implement the soundtrack for the game *Unsteady VR*, included with this thesis as a portfolio of compositions. Gameplay intensity measurements were conducted to ascertain the visual behaviour and interactivity of each level to create accompanying music. The measurements informed the composition and implementation of the soundtrack, ensuring playback of the accompanying music was aligned to the intensity of the game. While the Gameplay Intensity Framework was used to solve compositional obstacles specific to *Unsteady VR*, it is demonstrated as an adaptive tool capable of aiding designers in multiple contexts. This thesis elevates intensity as a governing structure for all stimulus presented in game, which composers and sound designers can reference to interweave their creations more deeply with neighbouring audiovisual assets.

References

Ludography

Arkane Lyon. *Deathloop*. Bethesda Softworks. PlayStation 5. 2021.

Bandai Namco Studios. *Ace Combat 7*: Skies Unknown. Bandai Namco Entertainment. PlayStation 4. 2019.

Beat Games. *Beat Saber*. Beat Games. Microsoft Windows. 2019.

Bethesda Game Studios. *Fallout 4*. Bethesda Softworks. Microsoft Windows. 2015.

Bethesda Game Studios. *The Elder Scrolls IV: Oblivion*. Bethesda Softworks. Windows. 2006.

Bethesda Game Studios. *The Elder Scrolls V: Skyrim*. Bethesda Softworks. Windows. 2011.

BioWare. *Mass Effect*. Microsoft Game Studios. Xbox 360. 2007.

Bohemia Interactive. *Arma 3*. Bohemia Interactive. Microsoft Windows. 2013.

Bungie. *Halo*: Combat Evolved. Microsoft Game Studios. Xbox. 2001.

Capcom. *Resident Evil*. Capcom. PlayStation. 1996.

CD Projekt Red. *Cyberpunk 2077*. CD Projekt. PlayStation 4. 2020.

CD Projekt Red. *The Witcher 3: Wild Hunt*. CD Projekt. PlayStation 4. 2015.

Creative Assembly. *Alien: Isolation*. Sega. PlayStation 3. 2014.

Cyan, Inc. *Myst*. Broderbund. 1993.

Daniel Mullins Games. *Inscryption*. Devolver Digital. Windows. 2021.

Developer Commentary (Left 4 Dead), accessed 12 December 2022, https://left4dead.fandom.com/wiki/Developer_Commentary_(Left_4_Dead).

DMA Design. *Grand Theft Auto*. ASC Games. MS-DOS. 1998.

Dotnod Entertainment. *Vampyr*. Focus Home Interactive. Microsoft Windows. 2018.

Downpour Interactive. *Onward*. Downpour Interactive. Windows. 2016.

Eidos-Montréal. *Deus Ex: Human Revolution*. Square Enix Europe. Microsoft Windows. 2011.

Epic Games. *Fortnite: Battle Royale*. Epic Games. PC. 2017.

Flippfly. *Race the Sun*. *Flippfly*. Microsoft Windows. 2013.

Harmonix. *Guitar Hero*. Red Octane. PlayStation 2. 2005.

Hello Games. *No Man's Sky*. Hello Games. Playstation 4. 2016.

id Software. *Doom*. Bethesda Softworks. Windows. 2016.

Infinity Ward. *Call of Duty*. Activision. Microsoft Windows. 2003.

Infinity Ward. *Call of Duty 4: Modern Warfare*. Activision. Microsoft Windows. 2007.

Infinity Ward. *Call of Duty: Modern Warfare 2*. Activision. Microsoft Windows. 2009.

Infocom. *Zork*. Personal Software. PDP-10. 1977.

Insomniac Games. *Spyro the Dragon*. Sony Computer Entertainment Games. PlayStation. 1998.

Interplay Productions. *Fallout*. Interplay Productions. MS-DOS. 1997.

IO Interactive. *Hitman*. Square Enix. Microsoft Windows. 2016.

Ion Storm. *Deus Ex*. Eidos Interactive. Microsoft Windows. 2000.

London Studio. *Singstar.* Sony Computer Entertainment Europe. PlayStation 2. 2004.

LucasArts. *Grim Fandango*. LucasArts. Windows. 1998.

Motive Studio. *Star Wars: Squadrons*. Electronic Arts. Microsoft Windows. 2020.

Namco. *Air Combat*. Namco. PlayStation. 1995.

Namco. *Pac-Man*. Midway. Arcade. 1980.

Naughty Dog. *The Last of Us*. Sony Computer Entertainment. PlayStation 3. 2013.

Naughty Dog. *Uncharted: Drake's Fortune*. Sony Computer Entertainment. PlayStation 3. 2007.

Nintendo EAD. *The Legend of Zelda*. Nintendo. 1986.

Playground Games. *Forza Horizon*. Microsoft Studios. Xbox 360. 2012.

Rare. *Banjo-Kazooie*. Nintendo. Nintendo 64. 1998.

Respawn Entertainment. *Titanfall 2*. Electronic Arts. Windows. 2016.

Rockstar North. *Grand Theft Auto: San Andreas*. Rockstar Games. PlayStation 2. 2004.

Santa Monica Studio. *God of War*. Sony Computer Entertainment. PlayStation 2. 2005.

Thatgamecompany. *Journey*. Sony Computer Entertainment. PlayStation 3. 2012.

Thekla, Inc. *The Witness*. Thekla, Inc. PlayStation 4. 2016.

Tripwire Interactive. *Maneater*. Tripwire Interactive. PlayStation 4. 2020.

Ubi Soft Montreal. *Tom Clancy's Splinter Cell*. Ubi Soft. Xbox. 2002.

Ubisoft Montreal. *Assassin's Creed*. Ubisoft. PlayStation 3. 2007.

Valve South. *Left 4 Dead*. Valve. Microsoft Windows. 2008.

WarpFrog. *Blade and Sorcery*. WarpFrog. Microsoft Windows. 2018.

Bibliography

Adams, Earnest, "Fundamentals of Game Design, 3rd ed", Pearson Education, 2014.

Bajakian, Clint (2007). ''Music: Design Production and Implementation.'' Paper presented at the annual Game Developers' Conference, San Francisco, March 4–9, 2007.

Boucher, Myriam and Jean Piché, "Sound/image relations in videomusic." In *Sound and Image : Aesthetics and Practices*, Edited by Andrew Knight-Hill. 1st edition. Abingdon, Oxon ;: Routledge, 2020.

Chion, Michel. "Audio-Vision: Sound on Screen". Edited by Claudia Gorbman. Columbia University Press, 2019.

Collins, Karen. *Game Sound : an Introduction to the History, Theory, and Practice of Video Game Music and Sound Design*. Cambridge, Massachusetts: MIT Press, 2008.

Ciesla, Robert. “Sound and Music for Games : the Basics of Digital Audio for Video Games”. Berkeley, CA: Apress L. P., 2022.

Lopez, Mike. “Gameplay Design Fundamentals: Gameplay Progression.” *Gamasutra*, 29 November, 2006. https://www.gamedeveloper.com/design/gameplay-design-fundamentals-gameplay-progression.

Lopez, Mike. “Gameplay Fundamentals Revisited, Part 2: Building a Pacing Structure.” *Gamasutra*, 26 November, 2008. https://www.gamedeveloper.com/design/gameplay-fundamentals-revisited-part-2-building-a-pacing-structure.

Hufschmitt, Aline, Stéphane Cardon, and Éric Jacopin. "Manipulating Player Performance via Music Tempo in Tetris." In *Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '20)*, 146-152. New York, NY, USA: Association for Computing Machinery, 2020.

Lopez, Mike. “Gameplay Fundamentals Revisited: Harnessed Pacing & Intensity.” *Gamasutra*, 12 November, 2008. https://www.gamedeveloper.com/design/gameplay-fundamentals-revisited-harnessed-pacing-intensity

Munday, Rod "Music in Video Games," in In *Music in the Digital Age*, edited by Jamie Sexton. Edinburgh: Edinburgh University Press, 2007: 61-63.

Peerdeman, Peter. “Sound and Music in Games by Peter Peerdeman.” (2010), *VU Amsterdam*

Whalen, Zach. "Case Study: Film Music vs. Video-Game Music: The Case of Silent Hill." In *Music in the Digital Age*, edited by Jamie Sexton. Edinburgh: Edinburgh University Press, 2007: 74-75.

Zdanowicz, Gina and Spencer Bambrick. “The Game Audio Strategy Guide: A Practical Course”. Taylor and Francis, 2019.