UNIVERSITY OF CALIFORNIA
SANTA CRUZ

GLITCHING GAMES STUDIES

A thesis submitted in partial satisfaction of the
requirements for the degree of

MASTER OF SCIENCE

in

COMPUTATIONAL MEDIA

by

Cassandra V. Ravenbrook

June 2023

The Thesis of Cassandra V. Ravenbrook
is approved:

Professor Nathan Altice, Chair

Professor Michael Mateas

Peter Biehl
Vice Provost and Dean of Graduate Studies

Table of Contents

List of Figures

Abstract

Glitching Games Studies

by

Cassandra V. Ravenbrook

Glitch in video games has been studied as an intentional aesthetic element and as an unintentional breakage that produced new narrative possibilities. Glitch art theory has examined interaction with glitch as phenomenological, but the examination of glitch as a systemic aesthetic in video games has gone under-explored. This thesis proposes that the defining power of glitch, in both cases, arises from the ways it makes the otherwise invisible protocols of computing visible in both domains and finds that both glitch art production and transformative modes of videogame play—such as speedrunning and narrative reparative play—have the capacity to make this glitch materiality productive.

To my amazing mother,

who taught me to never stop learning.

Acknowledgments

This thesis wouldn't have been possible without the support and guidance of my advisor Nathan Altice and his robust feedback. I would also like to thank Professor Michael Mateas for his guidance on many of its drafts.

The trip to this document was rocky, having started in Fall of 2020 and experiencing many false starts. I want to thank Professor Adam Smith for his attention in one of those; our chats helped me understand what a thesis could look like.

Also, thanks to the graduate students of OPEN and EIS Lab. All of you are doing kick-butt work, and without you as community, I wouldn't have made it this far.

Finally, I want to thank David Kant and for sharing with me his work on *ORGANVM PERCEPTVS*. That album didn't make it into the final analysis, but both the final result and the process behind its production were influential to the ideas I discuss here.

Chapter 1

Introduction

Glitch as an aesthetic is defined by processes of interruption within computational systems. While interruption and accident have been utilized and appropriated into the arts previously, such as in Duchamp's work (Erratum Musicale [11, p. 38] and The Bride Stripped Bare by Her Bachelors [27, p. 6-8], and in modern film and analog video works [27, p. 9-14], glitch aesthetic is unique in its reliance on the interruptions and accidents of computation. It relies both on accidental generation with complex electronic systems and means for that generation to be made tangible. Screens and speakers serve as interfaces that expose the presence of accidents, artifacts, and noise in ways that transform how we understand computation and the media which we consume digitally.

Digital interfaces make it possible to experience computational glitches which have spawned a vernacular of digital failure. These interfaces transform error as a conceptual element—as deviations from expected operation within systems—to material elements which can be experienced as representations. The accidents of glitch transform

how we can experience all digital art, including both pieces that are produced with digital tools or converted to digital formats. Formats provide opportunities for error in encoding, decoding, or transfer and spaces for surprising manipulations with or without specialized tools. Formats, their errors, and their subversions shape what glitch looks like, what glitching means as a methodology, and how we interpret glitch as a component of art, digital or otherwise.

Glitch art is defined by its utilization of hardware or software misuse or malfunction. The unifying aesthetic of glitch art is a sense of alienation from human intent through systematic error characterized by the noise, harsh color shifts, repetitions, and linear visual forms [36, p. 31] that glitch techniques generally produce . While strongly associated with these features, glitch art is not constrained to them. As a genre, glitch art is shaped by its origins in computational media and its methods shift with changes in the technologies it operates through.

Glitch art is often understood based on categories defined by how it is produced: glitches discovered by chance—captured and framed as art such as BEFLIX's early captures [47]; glitches induced within a system through hardware, software, or data modification such as Je Donaldson's circuit bending of the a NES console [35, p. 47]; glitch produced via the use of intentionally constructed glitching tools such as Glitch Codec Tutorial [14] l; and glitch as formal quality appropriated in an otherwise unglitched creative mode such as the paintings of Gerhard Richter [36, p. 25]. Despite the possibility of an audience indifference to the production method, these categories inform methodologies employed by glitch art practitioners. I will argue in this thesis that the methodologies of glitch art production are analogous to how player–users expe-

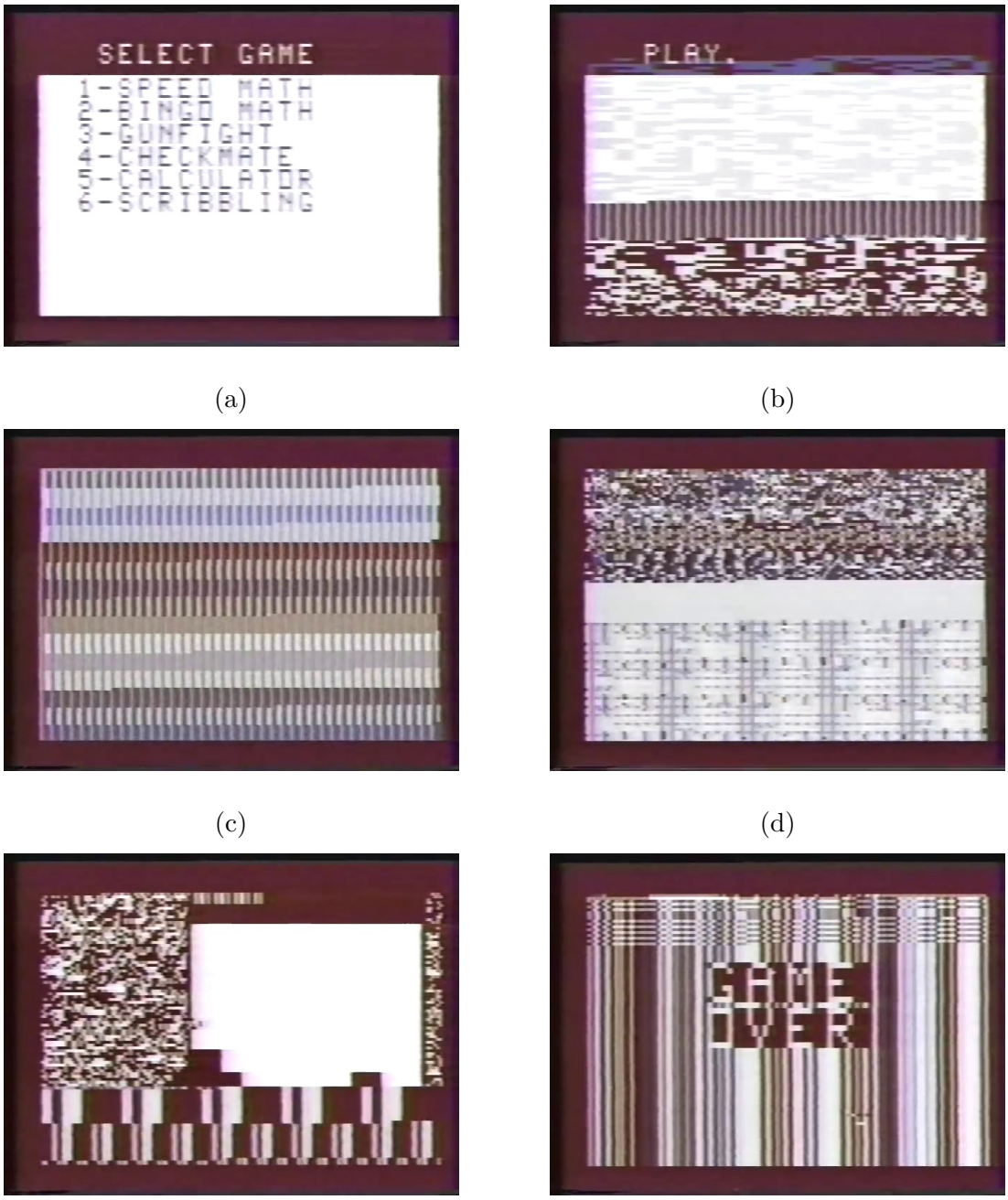
rience interactive glitch art when we extend glitch art theory into the realm of playable media.

Glitch art theory examines glitch and glitch inspired work to understand the experiences that glitch art produces and its potential as an arts methodology. Glitch lends itself to opportunities for change inherent in systems [35, p. 29-32], implicit misuse of tools [13], and queer expression [30]. As an academic field, Glitch art theory developed through the 2000's and 2010's. Kim Cascone examined how the glitch aesthetic developed in the 90's European techno scene; a result of the insertion of technological noise, a culture of sampling and remixing, and the introduction of digital signal processing tools used and misused by artists [16]. Michele White's "The Aesthetic Of Failure: Net Art Gone Wrong" from 2002 examines net art which implements or conveys its themes through breakdown [54]. These works by JODI and other net artists foreground the forms and limits of the net as a delivery medium. BEFLIX (Ant Scott) established a dedicated website and blog for documenting and discussing the aesthetic qualities of found glitches [47] and with Shay Moradi documented the common formal features of glitches through an arts framework [36]. In 2011 Rosa Menkman published *The Glitch Moment(um)* which formalized her theory of the genre after working in it for half a decade, positioning glitch art's critical potential as a potential embedded in technologies themselves. Carolyn Kane's analysis positions glitch art and its techniques within a lineage of modern artwork characterized by malfunction and accident [27] and identifies the aesthetic's ultimate ambivalence to the technology which enables it as both sign of breakage but also exploitable artistic technique [28, p. 125]. In *Glitch Art in Theory and Practice* Michael Betancourt critiques earlier theories' reliance on ma-

terial form. Rather than understanding glitch art as a result of a technical failure, he identifies in digital culture an aura of replication which he uses to frame the response to apparent error in glitch art. For Betancourt, glitch isn't glitch because of an error in the process, but only as the art is read as deviation from expected perfect replication, random, intentionally induced, or faked. [11, p. 95-100]. Betancourt ultimately affirms that without the correct framing, glitch carries with it no particular critical potential.

Jamie Fenton's *Digital TV Dinner* [41] is one of the earliest examples of glitch art, produced on the Bally Astrocade game system. Jamie Fenton was team manager of the development of this game console[11, p. 47], one of the programmers responsible for development of the system's on-board ROM [51], and the programmer of *Bally Basic* [20]. *Digital TV Dinner* is a video piece which produces its visual artifacts through the removal of game cartridges from the Bally Astrocade system during runtime (see Figure 1.1). Rather than halting, the system continues its operation reading through the instructions where the game ROM should be. Fenton discovered this glitch-producing technique during a dinner party while high on psychedelics, later collaborating with others to produce the recorded version for festival display [11, p. 20-31]. Digital TV Dinner is a formative example of glitch art because of its relatively early production, its visual aesthetic qualities which include elements which would characterize glitch as an aesthetic in the future (fragmentation, replication, linearity, and complexity [36, p. 28–32]), and the essential nature of interruption in its production which set it apart from the computer–video work it was displayed alongside [11, p. 29].

Images, music, and video are traditionally produced independent of input from an audience (or the input is interpreted and managed by an artist in the case of live



(a) (b)

(c) (d)

(e) (f)

Figure 1.1: Screen capture of *Digital TV Dinner* uploaded to YouTube in 2009 [41]

performance). This extends to glitch art as it is traditionally understood. Menkman understands noise, artifact, and breakdown within the context of sender–receiver models of communication [35, p. 12-14]. The glitch is found, induced, or invented and then employed as part of an object, performance, or transference and from there is received by an audience. From this framework we can explicate two relationships: an artist experimenting with a technical system or tool using glitch methods such as databending, datamoshing, glitch filters, hardware interruption, et cetera, and implementation of these techniques in creative production for an audience.

In "traditional" glitch art production, an artist experiments with a technical system in ways not explicitly intended by the object's or technology's designer, producing novel feedback which extends beyond the object's intended range of outputs. It is through this feedback cycle that the artist becomes attuned to the object and its behaviors and curates its output to produce static pieces or actively in live production. It is only after the process of experimentation and development that art or live performance techniques are presented to an audience (Figure 1.2). Shifting from an analysis of static or autonomous art objects to one of games or instruments requires reframing where the art and reception occur. For example, games are not complete art objects without the engagement of a player; their art arises in the agencies produced and experienced in play [38]. The art is in the relationships that unfold between the elements of the interactive media and the player. This relation is made clear in the problems that arise when trying to archive and exhibit playable media within museum contexts [7]. This difference in how the art is received fundamentally changes how we can understand glitch as an aesthetic across media. When players are a component of an interactive

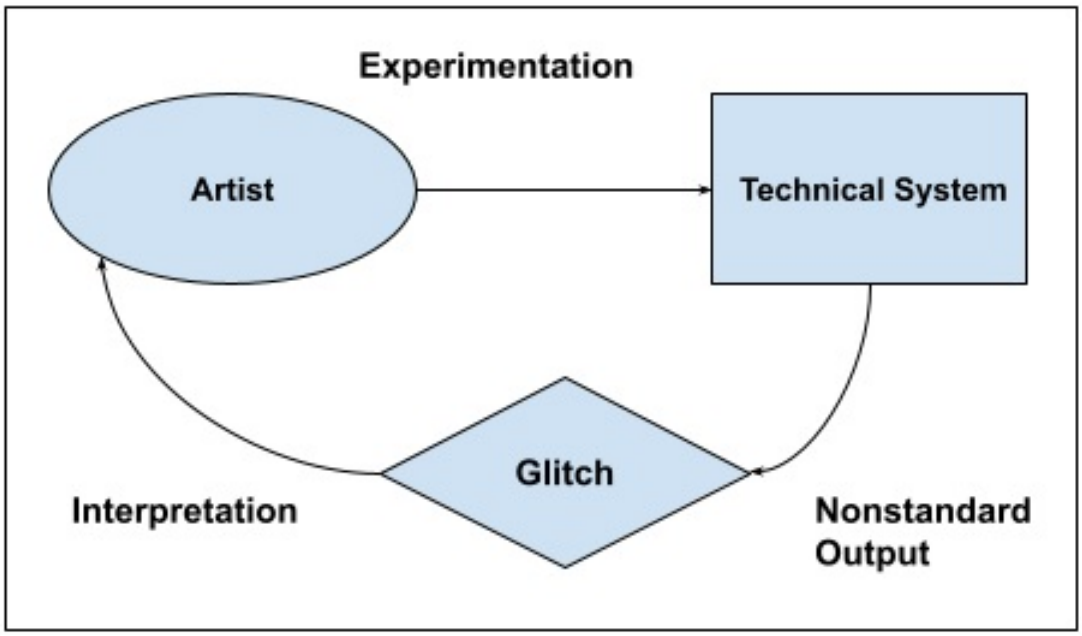


Figure 1.2: Glitch Artist-Technical System-Interpretation Loop

media system, the glitch becomes embodied. When glitches arise in these systems, they reveal similarities between glitch art process and systemic narrative play.

The glitch relationship that interactive media brings between players and objects is analogous to the relation between an artist and a technical system. The accidents of interaction and unexpected operation of the system transform the players' awareness and meaning–making within a game in the same way that experimentation with a complex digital tool produces new processes of meaning making and expression for an artist. This analogy allows us to formulate a new relationship where the glitch is not necessarily a relation between the artist and object but can arise also between an interactive object and the audience (Figure 1.3). We can now apply existing understandings of creative tool use to audience meaning making with interactive media to understand glitch as a relation in audience-driven interactive systems.

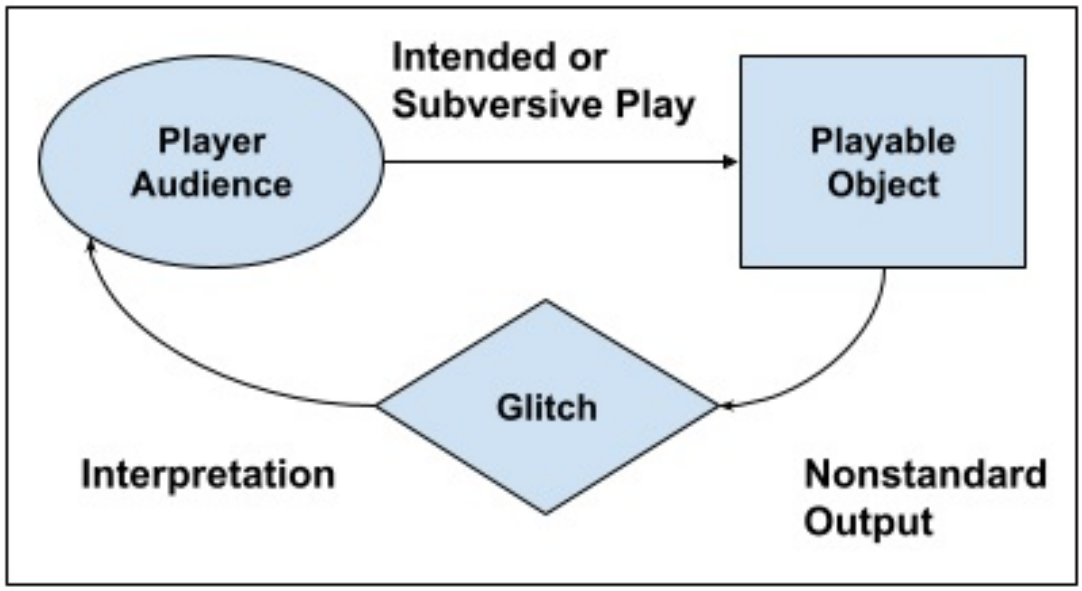


Figure 1.3: Player-Game-Interpretation Loop

In order to extend glitch art theory to interactive media I will be relying on phenomenology, the philosophical field concerned with describing human experience without reducing experience to analytical measurements. I will use phenomenological texts concerned with digital games and with queer experiences to expand how we understand meaning making in this context of audience play, experimentation, and systemic accident. I focus on games and game–like art objects/performances as they are spaces where glitches more often take on narrative and affective meanings. While frameworks for theater and literature interpretation have been applied in previous analysis of glitch in games [26, 23], more can be gleaned from using tools specifically tailored for unpacking interactive experiences.

Within games, despite ideally not existing or intended to not exist, glitches continue to be present and shape interaction and must be responded to. Play can continue, and grow in interesting ways, even in their presence. MissingNo., a glitched

creature in the original Pokémon Red and Pokémon Blue games for the Game Boy, is technically a result of the game misreading memory data that has the capacity to corrupt saved game data—destroying a tiny world—and forcing a complete restart. In competitive play, technical glitches are often turned into exploits [17, p. 114-115] that need to be policed or risk destroying the potentials for interesting play. This can take the form of intervention by developers as seen in competitive play [32, 49] or can occur democratically within speed running communities through categorical definitions [45] and meta-rules in fighting game tourneys [24, p. 117]. Menkman argues that the glitch was powerful because of the ways it could change our view of how media systems work, that after the uncanny moment of exposure, rather than being dismissed, glitches can ". . . instead force new knowledge about the glitch's techné. . ." (the operation of its underlying computational medium) [35, p. 31]. Games, as systemic experiences, are fertile ground for interrogating that type of experience.

Glitch art theory's notions of the interruption as a site for revelation can be extended through phenomenological understandings of interactive media. This thesis develops glitch materiality as a framework for examining glitch experiences in interactive media building upon embodiment as understood through phenomenology and touch, as contextualized by studies of human–technology interaction in within art spaces. I will begin by establishing the foundations of glitch art theory and then expand it by way of a cybernetic understanding of games and breakages. I will then examine through case-studies the analogous practices of glitch art production, technically, and narrative reparative play.

Chapter 2

Glitch Art Theory

While some glitch artists have sought to use glitch art as a component in critical arts practices, many glitch artists are affectionate rather than critical towards the technologies and systems they rely on in their craft. This can be seen in Ant Scott's (BEFLIX) production and curation of glitch pieces from 2001 to 2005 [47]. Ant Scott established the earliest website dedicated to computer "glitch art". From 2001 to 2005 he documented glitches generated using systems ranging from hardware-circuits to game emulators. His documentation and work position glitch art as a practice in producing, curating and understanding glitch as a method and a particular aesthetic. This can be seen in Scott's first post where he positions glitch art against fractal art. Ant Scott's curating of found and induced glitches is not critically motivated and his commentary tends to center around visual form.

Iman Moradi in *Glitch Aesthetics* distinguishes between the "Pure glitch" and the "Glitch-alike" [36, p. 4]. He defines the Pure Glitch as unpremeditated; it is a proper accident of the system. It may be captured and artful, but not necessarily so.

The Glitch-alike includes glitches intentionally triggered by artists or the recreation of glitch effects in non-digital mediums. This distinction between found and produced glitches and glitch effects is a recognized part of the glitch art discourse from Ant Scott's first 2001 post [47, 2001/07/12] through the beginning of the 2010's [15] and onward [11]. Moradi describes glitch art as a genre characterized by the presentation of glitch-alikes [36, 18]. Moradi's understanding centers glitch art around the visual reproduction and aestheticization of digital glitches in non-digital media. Moradi's aesthetic theory is likewise not concerned with glitch's potential for critical art. Like Ant Scott, his theorizing was focused on understanding the qualities and appeal of the object rather than the unique effects of glitch art as a medium.

Enter Rosa Menkman. Menkman is a prominent figure in glitch art theory from the mid-aughts to the early 2010s. In her defining theoretical text *The Glitch Moment(um)* she argues directly against the binary distinction Moradi makes between the glitch and glitch-alike [35, p. 36]. Menkman recognizes the distinction as a starting place but demands we go further and question why we are invested in such distinctions, urging us to explore the range of glitch art practices in their contexts. To Menkman, glitch art is about using glitch to shock and affect an audience. The titular moment(um) describes the tipping point when a breakdown is so jarring [35, p. 29-30] that it calls into question all the faith you had in the technological systems and forces you—with its momentum— to reevaluate how you engage with those systems

Menkman builds her glitch art theory on accidents in communication. Using Claude Shannon and Warren Weaver's information theory model (source → sender → signal (plus noise) → receiver → destination) [35, p. 12-28], she positions noise and

glitch as a technical occurrence enacted at the point of transmission, manifest as an accident. Menkman characterizes these accidents through Paul Virilio's understanding that a corollary accident is invented alongside any new technologies and that these accidents provide knowledge vital to understanding the world we live in [35, p. 32]. Noise artifacts, feedback artifacts, and glitch artifacts arise as unpredicted occurrences in communication technology. Menkman distinguishes between failure and glitch based on how the events are received:

> In short, failure is a phenomenon to overcome, while a glitch is incorporated further into technological or interpretive processes. Accordingly, when the glitch opens up to the realm of symbolic or metaphorical connotations, the interruption shifts from being a strictly informational or technological actuality, into a more complex post-procedural phenomenon to be reckoned with. [35, 27]

In contrast to Menkman, Betancourt argues in *Glitch Art in Theory and Practice* [11] that glitch art fails to achieve its critique of technology. Taking a critical theory approach, he anchors glitch art in earlier practices engaging with breakage and medium-driven practices. He argues against an ontological understanding of glitch art justified by its materiality. In his critique of ontological strains of glitch art theory such as Menkman's, Betancourt develops the concept of the *aura of the digital* to show how glitch, as theorized, fails to function as a critical medium. The aura of the digital refers to how we read and understand digital information [11, p. 92]. For example an audience could be presented with three digital images apparently identical, but encoded and compressed in different formats. Within the aura of the digital an audience would assume that each one could refer to the digital instance because they have, in seeing the image, received access to them and they seem to be the same.

Glitch art, as digitally mediated and produced, also is experienced within this aura [11, 92-93]. The apparent presence of data corruption as indicated by artifacting can be witnessed when files are rendered with systems that reveal those artifacts or in data where the outcome of that rendering has been saved. As a form of generated art, glitch artists cede agency to accidents of mutation and decompression. This framing argues that by examining the visual glitches, we don't necessarily come closer to understanding the digital representation of the image. The image artifacts are not caused by an error in calculation but are merely the logical result of data—unexpected by the software's designers—being processed [11, 96-98]. Betancourt thus argues that, rather than a glitch revealing some deeper truth about the underlying materiality of the system (its data and algorithms), the understanding of an artifact as a glitch resides in its interpretation.

To overcome the aestheticization of glitch and produce a critical piece, the breakdown can not just be invoked as a sign but must be integrated into the work. Through case studies of breakdown in *Halt and Catch Fire*'s (2014) opening sequence and Bill Morrison's *Decasia*, Betancourt distinguishes between non-critical and critical employment of glitch. As of 2014, glitch as a semiotic shorthand for technological breakdown has been integrated into the common vernacular of film. The blocky artifacting effect seen in *Halt and Catch Fire*, (Figure 2.1), and its origin as a simple procedure in photoshop, is an example of how the glitch aesthetic serves as a metaphor of breakdown (in theme with the show) but does not actually interrupt or create a stoppage within the audience experience [11, p. 108-112].

Decasia is a documentary constructed from aged film, bearing corruption from

Figure 2.1: Screenshot of *Halt and Catch Fire*'s (2014) credit sequence

chemical decay. The age exposes the materiality of its source and Morrison integrates the rot itself as an entity in the film. The decay is not only a signifier of decay in the film itself—but is made core to the reality of the reality we are watching the film from [11, 112-122]. The rotting of the film points to its status of an object constructed from aged materials and when film rot obscures its figures, see Figure 2.2, it demands we associate it with historical video technologies and their limitations.

The question both Menkman and Betancourt were attempting to address was the ossification of glitch art in the late aughts and early 2010's; a genre whose ethos was often founded on surprise and breakdown. Hugh Manon and Daniel Temkin compare this with the shift in surrealism, tying it directly to the influx of images being produced as the style grows in popularity [31]. Rather than a smaller clique of artists focused on pushing the genre in new directions, the average product of glitch art comes to

Figure 2.2: Screen from *Decasia* (2003)

rehash the same production techniques time after time producing works recognizable as glitch but ultimately in line with glitch as a mainstreamed aesthetic with predictable results. Manon and Temkin in their notes point to how Ben Baker-Smith's "GlitchBot" [8], which automatically scrapes creative commons images on flickr and glitches them, parodies this state of affairs— it's most disarming when it posts an image that isn't glitched at all.

In 2010 Menkman released *A Vernacular of File Formats* [34] wherein she rallies against the commodification of glitch in the form of the now stereotyped filters, coffee table books, Kanye West music videos, and nerdcore subculture. The strategy with the exhibition and glitching guide was to make the more common types of file glitching techniques transparent and understandable and therefore passé. Menkman also designed a glitching tool Monglot (programmed by Johan Larsby) [35, p. 8]. Mon-

glot theoretically skirts the line between use and misuse. Rather than modify and interact with an image through layers of raw RGBA values or explicitly provided filters, Monglot's visual buffer is not directly editable and only shows the decompressed form of the working file. Rather than a palette of tools, Monglot provides methods to randomly modify data within the file as compressed. The visual buffer then refreshes and displays the file after decompressing it again [33]. Monglot demonstrates a possible path to try to move nascent glitch artists away from using simple glitch filters towards a practice of technical understanding of standards and their implications.

Nick Briz—who worked in the same local glitch art scene as Menkman—dealt with the popularization of glitch by shifting his understanding of it from style to methodology, best understood through his arts and education practices. Nick Briz's "THOUGHTS ON GLITCH[ART]v2.0" conceptualizes "glitch art" as the practiced misuse of a tool: ". . .it's about consciously doing things the wrong way." [13, p. 6] Using a tool to generate apparent glitches may lead to good and interesting results, but doing so is not practicing glitch art. Like Menkman, Briz believes that glitch has unique critical potential, however he centers glitch art as a practice for digital literacy rather than as a quality or materiality in the context of art. Briz refers to this framework as a glitch ethic.

Briz uses the work of Laimonas Zakas, posting under the name Glitchr, to exemplify the glitch ethic. In the early 2010's. Zakas experimented with nonstandard characters and tags in Facebook posts creating a recursive onion of weird effects that would render for anyone who saw the post on their timeline. [44] To understand the anomalies and recognize them as craft requires at least a passing literacy in how the

web renders its content.

Briz's own Glitch Codec Tutorial is a "a tutorial/essay on the technical, theoretical, and critical process of glitch art" [14, p. 2]. This combination guide and Linux ISO file[14, 6], provided pre-burned to a boot disc in workshops, puts tools for creative misuse into untrained hands. The boot disk includes a hacked video compression codec that enables its users to perform datamoshing—a particular glitch technique that eliminates or replaces key image frames in mpeg video allowing motion-based frames to be applied to images that they weren't intended to. This misapplication of movement data is what generates datamosh artifacts in the video.

By framing glitch art through a framework of technological literacy rather than anchoring it in some affective power, as Menkman does, Briz addresses a problem of audience that has troubled glitch art since its early theorization. In 2002, Michele White's primary questions with net art pieces that relied on technical breakage and failure in their deployment was their effectiveness in provocation [54]. An art literate audience can only pull so much meaning from the objects without also having the technical literacy to understand how they are breaking. However, a systems expert who can diagnose the breakage will certainly be less shocked and affected by it. Briz's glitch ethic serves as an explicit invitation to learn, not just descriptive forms, but the methodologies and technical operations used to invoke those chaotic results.

Glitch art theory revels in glitch art as a creative mode that centers creators' experimentation with their tools rather than operating as a coherent theory of reception. This framing allows us to focus on the interactive loops of glitch art production and the development of technical mastery in glitch as a set of creative techniques which

exploit creative potential through misuse of designed software and the development of purpose built tools such as: Menkman's Monglot[33], Briz's *Glitch Codec Tutorial*[14], Kim Asendorf's ASDFPixelSort [6], and Paul Hertz's GlitchSort [25].

This explication of glitch art theory discussed how systems are compromised and misused for new types of creative potential. In the next chapter we turn towards how interactions with the interfaces and systems of games produce meaning. This will prepare us to understand how new meanings are produced through the breakdowns in those systems.

Chapter 3

Mechanisms of Meaning Making in Games

In digital tools, computer mediation, and video games, glitch can uniquely arise as jank, disruption in a player's sense of control over game entities. These breakdowns are experienced in the gaps between users and digital systems. They are read as failures of the technology to seamlessly extend a user's agency as a result of errors within the system or imperfect expectation. This gap, and the experiences within it, are best approached with frameworks purpose-made to unpack how humans experience the world through their body and tools. To this end we turn to phenomenology.

Phenomenology is the study of being in the world. At its core are the *phenomena*, our perception of objects in the world through their effects [PtB 9]. Phenomenology provides frameworks to understand ourselves as embodied subjects and how, through habit, tools come to extend our body and how our use of them becomes preconscious. [29, p. 23-25]. Tools exist contingent to the body. When a tool functions as expected it

escapes our notice as an object in itself. This notion is called *ready-to-hand* [55, p. 74]. When the tool fails we come to contend with it based on what it "should" do for us [2, p. 48]. This state is called *present-at-hand*.

What tools we use to perform a task is determined by what tools we know how to use, which is a result of what tools have been made available to us in the past [2, p. 56-57]. These understandings shape how we approach any new tools in the future. These entrained understandings are called embodied literacies [29, p. 77]. Embodied literacies are not value neutral. They carry a host of values about what results are valuable. These values are inherited as literacies are passed to new tools informed by the literacies of the old [29, p. 80] and determine which tools and orientations towards tools are legitimated as normal [2, p. 91-92].

Seamlessness is experienced as a result of particular embodied literacies effectively orienting a user towards systems in ways specifically designed to go unquestioned. It is a relation between our bodies and the technologies we use [4, p. 87-88]. Seams are locations of touch between bodies within a system, sometimes physically, as in the interface between the users and controller, and sometimes figuratively, such as the collision between game entities in a constructed world. Noticed or unnoticed, seams serve as locations of meaning making and the construction of agencies [4, p. 90]. They can be experienced both as a connection to the system that enables control and a gap which reveals difference.

In games, glitch occurs when embodied literacies, which shape both how we interact with and how we read the seams of systems, fail. Phenomenology allows us to read this failure not of the system, but between the system and the user as a "failed

orientation: a tool is used by a body for which it was not intended, or a body uses a tool that does not extend its capacity for action." [2, p. 51]. It is here I will be constructing *glitch materiality* as a theoretical framework, invested in how glitch and hacking make and break meanings at the line of the seam and how the system contexts which choose, construct, and discipline action around seams respond to those breakages.

Because games are often understood as systems for designing agencies [38, p. ??], games studies provide a robust literature on the experience of interruption of agency and failure. In this domain, glitch-like experiences as "failed orientation" has been approached by the literature in two forms: Glitch as an effect on a player by the game, which I will discuss in the form of jank, and glitch as an employment of unintended technique of the player on the game, which I will contextualize through speedrunning as a practice. Through these two modes glitch creates new possibilities for negotiated and adversarial approaches to interacting with our technology and tools.

Extending glitch theory to interactive digital media requires that we not engage with it merely as an output of a system, but as a phenomenon experienced within the system and between its component parts. Brendan Keogh's *A Play of Bodies* describes the experience of playing video games as ". . . a play of bodies that dances across actual and virtual spaces." [29, p. 48]. Keogh's phenomenological approach sees both players and games as bodies that possess agencies that wax and wane through the activity of play [29, p. 40]. Keogh argues that to effectively read meaning we need to understand them as embodied textualities, texts that aren't bound to one object but dispersed across bodies within a system. In Keogh's model glitch becomes an aspect of embodied texts requiring an embodied approach to interpretation.

Keogh uses cybernetics to frame the players as "situated and embodied subjects". [29, p. 39–40] This approach positions the player as one body among many and anchored as a body in the world rather than as a purely cognitive mind interpreting without concern for bodily experience. [29, p. 23-24] Keogh leans on Hayle's understanding of posthumanism to support an embodied approach to engaging with games as text. Posthumanism recenters the role of embodiment in our relationship to information and knowledge, positioning the body as caught up in the technology of information and not transcendent over it [29, p. 28-30]. Keogh understands video games as intermediative rather than interactive [29, p. 40]. A player does not act on a game and is acted on in return as much as they are mediated by the game in their expressions of agency as they game is mediated similarly by the player. It is in this technical mediation Keogh understands the relationship between players and games as cybernetic.

A player's experience of play is always mediated by the body that is engaging with the game. We may forget hands and controllers as we are engaging, but that does not mean they cease to be channels for intent within the system. Neither can we forget how feedback moves from the game to the eyes or ears of the player. In order for play to be successful, both the systems of a player and the game need to be operating inline with each other. In discussing how technology extends our bodies Schmalzer notes that in order to connect to a character a player's ". . .influence needs to seamlessly extend into that game world." [45, p. 176].

Seamlessness, when engaging with technology, comes to indicate some presumed normal state of affairs, where intent flows from user into tool without interruption. A seamless experience is perhaps best understood by its opposite. When I first

tried to engage with *Garry's Mod* (2006), a creative interface and physics sandbox for Valve's Source engine packaged with a large collection of assets [37], I bounced off of it pretty hard. I hadn't had experience with similar tools or deep experience with any of the games that ran on the Source engine. The shelves of objects and characters presented to me meant nothing and I struggled to instantiate an object into the world space presented. The interface was covered in tools that I didn't know how to select for use. My lack of skill with the tool shoved me out of the use context into the world of desperately mashing keys to see if any of them did anything I could comprehend. The object failed to be ready-to-hand, but was too complicated for me to easily step back and speculate around what it should have enabled me to do.

Seamlessness is an aesthetic which flows from the convenient control of technology.. "Essentially, this aspiration [of seamlessness] aims towards experiences where people are no longer aware of the technology, the interface or the differences between human–technology and human–human interaction."[4, p. 87] It is about the ease of everything just being ready-to-hand.

Anne Cranny-Francis' in her examination of wearable technologies has proposed seamfulness and semefulness as related terms to help us consider not the ease of control over technology, but how meaning is produced through where interaction occurs. A seam is a material location of access and interaction. [18, p. 33-34]. A seme.is a location of meaning made through interaction with a sign [18, p. 35-36]. Both concepts are intended to foreground the constructed nature of our technologies and our interactions with them and complicate the presumption that seamless experiences are ideal..

The seme has potential as a way of understanding the interactions players

perform while interfacing with games. *Crusader Kings II* is a grand strategy game where the player controls subsequent rulers of a medieval European kingdom characterized by dozens of interlocking systems. For example your ability to parse the relationship networks will affect how effectively your ability to build alliances is vital for safety. Likewise to start a war itself requires forms of casus belli which are heavily dependent on who your ruler is married to, who they hold in their court, their religion, the current state of the papacy et cetera. A semeful interface that contextualizes each point of connection is required to naturalize the literacies that allow meaningful play.

When you move your mouse across the map, each region, city, or army opens up a tooltip description. This provides a truncated summary of the entities features and indicates a point of access into that object with a context specific menu for information and interaction. If it is a region, it will open up a window with status and a portrait of its current holder—another body relative to you—which glows and presents a tooltip, Figure 3.1-a and can be touched opening up their status window. This opens up another interface showing how characters are connected to their family and abstract entities like a faith which informs your possible moves in relation to them, (Figure 3.1-b). Built on their semes we can understand games like *Crusader Kings II* to be about the examination of their presence and the theorizing of operations within them. The systems are ever brought to the foreground and demand that meaning is made there.

Games designed to be seamless will always prioritize the embodied literacies already possessed by players. They will be designed to place the player in a constant state of ready-to-hand and in doing so deny scrutiny as systemic objects. Ahmed finds a similar vanishing for classic phenomenology, “For Husserl, then to see the table means

(a)

(b)

Figure 3.1: Semes in *Crusader Kings II* (2012)

to lose sight of its function." [2, p. 35] The position where we lose sight of a game is privileged by both the histories which place us in proximity to games as objects allowing us to learn them through repetition. Seamlessness exists as an aesthetic experience not in an object; it arises from internalized, embodied knowledge of how to deal with seams, where we have access and where we read meanings.

The embodied literacies of controllers, as discussed by Keogh, serve as a key example of this type of training .Embodied literacies are ". . . where the player learns how to perceive and embody the videogame through particular gestures trained by particular input devices." [29, p. 77] Keogh locates embodied literacies at the input device, whether it be a controller or QWERTY keyboard layout [29, p. 78-79]. This concept can be extended through the various levels of interfacing in computer software. Users can learn both how the baseline operations of a system works and understand how their inputs are structured to provide or block controls over certain parts of that system.

Embodied literacies prefigure what types of games we play. As the medium for enabling the learning and employment of new types of agency [38, p. 25-26], games affect how we understand how we can exist in the world. Video games, in building complexity through and on top of embodied literacies, prefigure certain types of agency. Through our agencies different objects put themselves within reach. The embodied literacy that comes with the ready-to-handedness of a hammer can open our eyes to our own control and agency of our bodies and in the world [45, p. 181]. Glitches draw attention to seams that our embodied literacies have made invisible, revealing our capacity to be shaped by the assumptions of those literacies and how we can choose other agencies for ourselves.

While seamlessness through embodied literacies is a common strategy, many games have built themselves on subverting the presumptions of convenient control. Jank, as an embodied experience, runs counter to seamlessness in many ways. Jank[46] in its simplest form is the occurrence of interruptions within the cybernetic circuit made up between a player and the media; it is a haptic aesthetic, but can be triggered from anywhere in the feedback loop between game and player. Jank can be intentionally employed to reveal seams and queer our interactions with games.

In "Janky Controls and Embodied Play: Disrupting the Cybernetic Gameplay Circuit", Schmalzer discusses two modes of intentionally employing jank in ways that run counter to ideals of control in games.Schmalzer uses Bennett Foddy's *QWOP* as a case study for intentionally exclusionary jank [46]. Rather than a smooth animated motion and climbing, like what is seen in Ubisoft's *Assassin's Creed* games where the main character automatically runs and climbs as long as a button is held, in *QWOP* the four control keys–Q, W, O, and P– are tied to forces acting on a ragdoll runner's muscles. This arrangement forces the player to think of their character as a complex physics object. Both the runner's feet can catch on the track and serve as a fulcrum, stumbling or backflipping runner into the ground, forcing a restart. In creating janky controls via the method of enforcing a mode of interaction in which the player has no literacy, QWOP shows how:

> Janky controls call into question the default subject position by undercutting the assumption that players are always able to accurately manipulate on-screen action through physical inputs that meaningfully translate into the game-states they desire. This reveals that players are only able to display mastery over games because they have been disciplined by games to behave in certain ways while games have been designed for their bodies. [46]

Another intentional use of janky controls examined by Schmalzer is the creation of empathic connection between player and character. In *Octodad*, behaviors like traversal across space and picking up objects are made complicated through how the controls move Octodad's body. In most games where a player controls an on screen character, players orient an analogue stick which then moves a character in reference to the frame of the game's camera. Movement animations visually normalize this movement. Intent is neatly translated and the visual feedback in line with how bodies are expected to move in the world. In *Octodad* the player controls their character's legs independently with each analog stick. Moving the entire character requires pulling him along, leg over leg. The controllability here becomes another realm of representation within the cybernetic circuit signaling a shift in state or a state perpetually outside the norm.

Anable's aesthetic of failure can also help us understand notions of interruption against the dominant contextualization of games as seamless experiences guiding us to success[[3, p. 104-105]. The games of Pippin Barr and Messhof are made case-studies for this aesthetic of failure.

Pippin Barr's *Let's Play: Ancient Greek Punishments* (2011) and *Snek* (2013) engage with cruel optimism and failed orientations within space. *Let's Play* features several minigames that all place the player in situations where success is made impossible in various ways. The primary mechanic is the clicking of the mouse rapidly, but this interaction takes on different meanings: in "Sisyphus" it is the means through you push your stone up the hill; in "Prometheus" it is the only means to keep an eagle from devouring your liver; in "Zeno" it is how you get move forward. Whether you inevitably

drop the boulder upon reaching the top, can't maintain the struggle and lose your liver (again), or seem to be able to progress, but never pass "halfway there", the seam that fails is not one in the game but in the endurance of the player themself.

Snek is an adaption of *Snake* for the iPhone, but with an intentional rejection of the design principles of what makes a "proper iPhone game" [3, p. 113]; rather than gentle swipes, this adaptation of Snake requires violent twisting or thrusting, betraying expectations of smoothness and containment. Snek extends jank into the space around the player demanding grand and wild moves to control. Itself is an intentional failure when contextualized within the design ethos of its platform. These games, Anable argues, in asking us as players to perform failure and ". . .disturb our assumptions about how games are supposed to work and how our bodies have been trained to interact with them." [3, p. 114]

Anable uses Messhof's games *Pipedreamz* and *Nidhogg* to present us with a different way of failing. *Pipedreamz* is a two-phase Flash game which alternates between menial burger flipping with *Red Light, Green Light* clandestine beef eating and technically challenging trick surfing. *Nidhogg* is a semi-round based fighting game where bouts are made asymmetrical via stage hazards. Both feature pixelated worlds with heavily abstracted figures underscored by tight mechanical interactions. The abstracted representations block players from fully contending with the mechanical complexities of control in the game: "What comes radically undone in the relationship between the visuals and the mechanics are the expectations of fairness and consistency that undergird most digital gameplay." [3, p. 119] *Pipedreamz* uses its easy to master menial, fast-food labor to anchor and contrast the complex surfing mechanics in its partner minigame.

[3, p. 116-117]. In play the game alternates between these modes as the player succeeds and fails over and over again. Anable points to all of these games as mechanisms for challenging how we understand failure. Through them we can "become attentive to the relationships among our failings, our feelings, and the systems with which we interact." [3, p. 120] Rather than understanding failure through a cruel optimism that demands we just keep trying or a shame which configures ourselves as the issue, these games construct spaces to understand failing within a larger context and ask us how we feel about it.

Play can exist beyond the agencies that are designed for players. Intentional jank and in-built failure may move against dominant trends in the industry, but still center the authority of games' creators. There are practices through which players impose their own meanings on these produced systems. One of them is the community-oriented practice of speedrunning.

Glitches, within the systems of a game, can often be exploited by players seeking to subvert the planned order of play. Speedrunning is a practice which often uses this type of glitch to finish games in record time and under. To that end, players construct whole new sets of embodied literacies for this self managed form of striving play[45, p. 184]. Quoting from game designer Bennet Foddy, Schmalzer compares speedrunning as becoming an expert in the design and systems of a game, admiring it with great respect, and then breaking the game "over their knee" [45, p. 92] Embodied literacies repeated across the histories of games go to speedrunning to die. Practicing speedrunners are apathetic to the affective or mechanical semes as presented by the developers instead investing meaning in the evolution of the routes and tech they practice. New

semes arise in the technical accidents which can be exploited in the name of speed rather than any narrative diegesis [45, p. 160].

The smashing of the game in the construction of a speedrun route is also a construction of a new mode of being [45, p. 92-93]. Schmalzer finds the potential to change the rules note only in speedrunning as a practice, but in the communities that build up around the games and the rules they produce for themselves. [45, p. 100] "Categories are only useful if they create possibilities for those that are interpolated into them. And new categories become glitches that can change our orientations towards media, technology, and culture." [45, p. 105] Schmalzer sees in speedrunning potentials to construct new spaces, new agencies, and new ways of being embodied according to individual desire. However, new orientations with games can occur outside realms of deconstructive play, even being intentionally produced.

If, foundationally, games are texts which create their meanings through the mediating play between the body of the player and the bodies of a game's space, with the interface serving as foundational points for the shaping of experiences, than aesthetics of jank and failure can come to reveal a materiality to the form, cracking cultural ideas of transparent reading and seamless control. Some experiential breakages can be produced accidentally and intentionally; however I see the possibility for an injection of dramaturgical agency in interactive media that goes beyond just the affordance of interaction, to be discussed in Chapter 5. If the death of the author [10] allows us to engage with texts disjoint from their producers, understanding interactive play through the lens of glitch art production, what I am calling *glitch materiality*, lets us read acts of play in digital interactive texts that go beyond their intended mechanical ontologies.

Interactive bodies are not left complete by authors, they must be completed and read during and after engaging. Reading them means watching how this body left behind by the author can be fabricated with the body of the player into a dramaturgical Gundam of meaning. Before we arrive there however, we must once again return to the discussions of glitch art production and how intentional interference can reveal digital aesthetics not meant for human eyes.

Chapter 4

Technical Glitch Practice

Beneath every digital standard there exists expectations about how that standard is maintained. It is the goal of professional software and its intended use to hide those standards behind interface so a user can perform their work quickly without having to consider the low level operations of the computer. One of the operations of glitch art is to reveal those standards through interruption and misuse and often produces effects which are not oriented towards human perception.

To demonstrate this operation, this chapter uses databending—the use of unintended tools in modifying file data producing artifacts in those files when reinterpreted—to demonstrate glitch as a systems oriented practice. Databending techniques rely on the capacity for file formats to fail catastrophically, but incompletely and instead express their underlying algorithms in unintended ways. It is the combination of these breakdowns and human intervention and interpretation that we identify the act of glitch art [Betancourt15 37-39]. It is in the acts of interruption and interpretation that glitch artists are motivated to design for and curate particular effects.

The technical system I will be engaging with through this case study is the PNG file format, a lossless image compression format released in 1996. This standard can be repurposed to produce a variety of effects through a collection of custom Python scripts. I will examine the technical aspects of the PNG format and how these aspects lead to particular visual forms when interrupted. I will explain the operations of the format and the effects as the results of a system and gesture to their visual form. These forms are emergent and non-intended in the design of the file format; the consequences of decisions made in pursuit of a particular set of goals—lossless compressibility and speed of transmission. The goal of this case study is to reveal the relationship between the interference of a glitch artist, the cascading effect of the system running as-normal, and the technical master which develops in the process of causing instrumental operations gone awry.

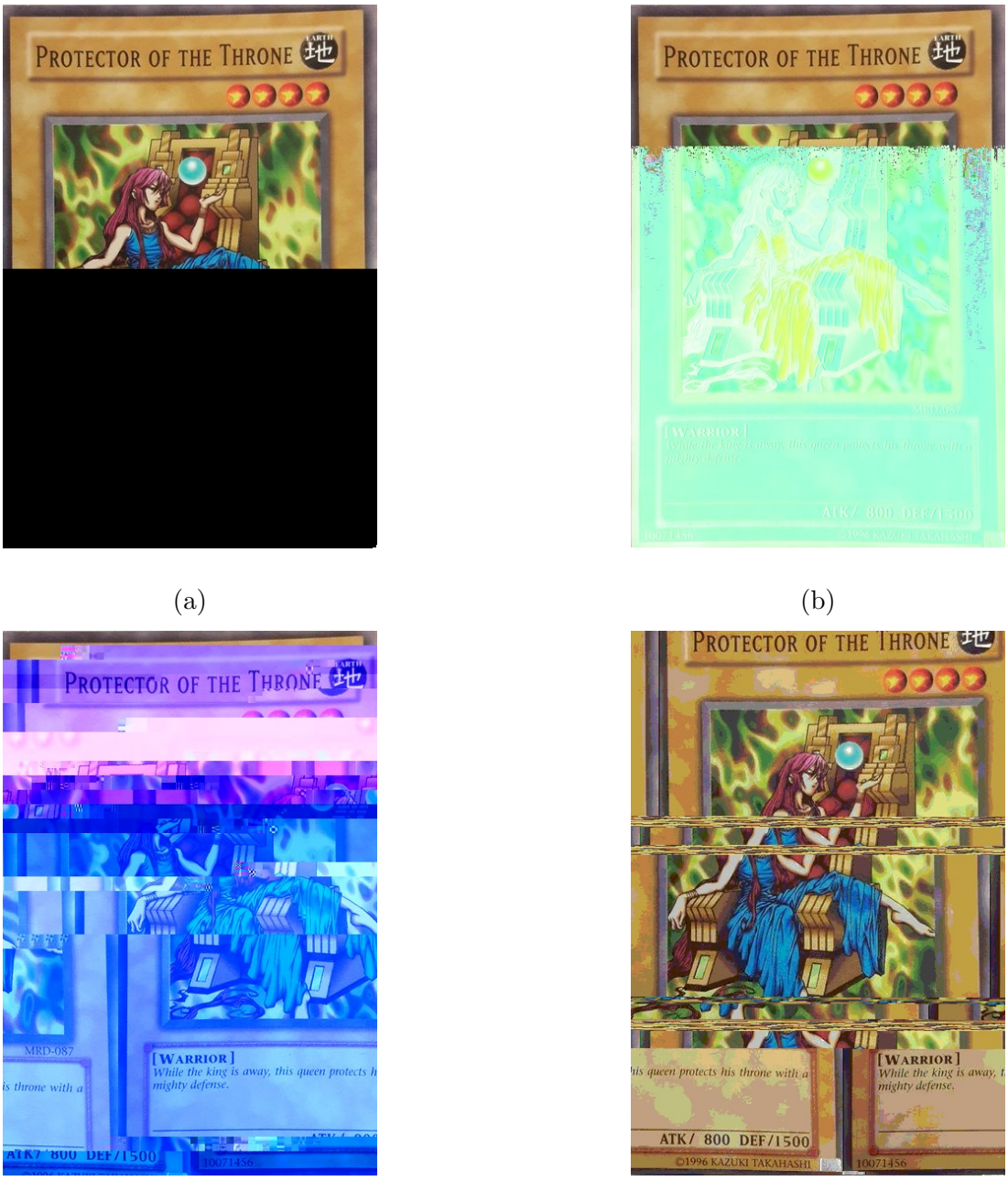
I examine the PNG file format because of its relative lack of formal examination compared to the JPEG (Joint Photographic Experts Group) format. JPEG is likely the most widely used image format as a result of its radically effective file compression scheme, even while both PNG and JPEG are prominent standards [21]. Glitch artists in the 2000's scene also have already closely exploited/examined the affordances of glitching the JPEG form [5] [34].

Databending is the practice of modifying a file using an application not designed to provide a human-friendly representation of its file type nor provide modification affordances for files of its type. One of the most common and accessible databending techniques for generating glitch artifacts in image files is opening them in text-editors, and manipulating their text as represented by text character encodings.The effective-

ness of this technique is heavily dependent on the encoding scheme of the file and favors certain file types over others.When this strategy is applied without care to PNG files, for instance, rather than generating interesting artifacts, the modified file often just halts its rendering process part way through.

PNG files are comparatively delicate in this regard. The format encodes and compresses image data into a series of data blocks each marked with a dedicated length, chunk label, data portion, and checksum. In PNGs, compressed pixel data is stored in IDAT (image data) blocks [42]. During rendering there are several ways that a databent PNG can cause an error on load: a change in the length of the data in an IDAT block can lead the decoder to hit a section of data where a new block label (e.g. IDAT, pHys, iTXt) is expected but not found, leading to a halt; a decoder may also halt the display of a PNG when its decompression implementation throws an error (as observed in modern web browsers); or when it attempts to validate an invalid checksum value. This common halting behavior, regardless of how it's triggered, means we need more care and understanding when compared to JPEG and BMP (bitmap) formats (Figure 4.1). Neither JPEG nor BMP images have built-in checksum functions that are tested during decompression that scrutinize for improper modifications. However, as a format designed to be both lossless, compressible, and interlace-able, PNGs glitch in ways neither of those formats will.

PNG compression utilizes Huffman coding during compression. In Huffman coding, the most common values are encoded into smaller representations— and the more homogenous the data is the greater the efficacy of the compression. If a program can losslessly map its data from one form to another more homogenous form, pixels



(a) (b)

(c) (d)

Figure 4.1: Text-editor based databending approach on common image formats

a: PNG Interrupted by a single modified byte several data-blocks into the file. (Google Chrome Version 109.0.5414.121).; b: PNG file interrupted by a single modified byte re-rendered in Paint (Windows 10) ; c: JPEG image response to several deletions and insertions; d: BMP image with similar deletions and insertions.

in PNG's case, then it can be more effectively compressed. The PNG specification[42] offers four mapping operations which can be invoked on a line-by-line basis based on a flag byte, (the first of each row). These operations are: Sub, Up, Average, and Paeth; each of these maps bytes of the image from color space to a color difference space, based on one or more of its neighbors, (in the same color channel). Rather than data being compressed as representations of RGB pixel values, data is compressed as a difference[1]. If an image is characterized by a lot of consistent change in the red, green, blue, or alpha channels, (or value in the case of grayscale), prefiltering will produce a more homogenous, and therefore compressible, representation.

Figure 4.2 visual shows this in action. Figure 4.2-a shows the bytes value of a prospective image before pre-filtering, sans the filter bytes themselves. Figure 4.2-b shows the same image with the SUB prefilter applied. In this near ideal the Figure 4.2-a features a regular gradient with a constant difference between columns. This constant difference characterizes the constant values outputted from the prefilter.

We can understand each scheme through the directional difference relationship of each mapping. Sub maps the difference-relationships across rows; see Figure 4.2. Up maps them across columns, Average maps the average difference between a pixel and its up and left neighbor pixels. Paeth chooses the smallest difference between vertical, horizontal, and diagonal neighboring pixels for each pixel encoded. Figure 4.2 also shows how, at this point, the data stored has been abstracted a step from what a human would be expected—from color to difference—to see or interpret. The prefilter process opens

[1] A mathematical analogy for this: if we consider each the lines, columns, or diagonal paths through a PNG as analogous as the expression of some mathematical function, rather than a sequence of data points only, the pre-filter provides us with a version which would be analogous to the derivative of that function.

(a) (b)

Figure 4.2: Near Ideal Application of PNG Pre-filter

Representations of the underlying data of a simple grayscale gradient with no PNG pre compression filter and with the Sub(traction) filter. a: A grayscale gradient 16 pixels wide with a change in value of 16 each column, it begins at a value of 144 and overflows to 0 when reaching 256 Without any pre-compression filter this data contains 16 unique values for its pixels. b: the same data with the Sub filter applied. This image is defined by a column of gray (0x90-value pixels), which match the first row of the left image, followed by homogenous rows of dark gray (0x10-value pixels) which map the differences in the previous image. The pre-compression filter makes the data more homogenous and leads to more efficient compression.

up the possibility for modifications to create cascades of unexpected values. The look of these cascades depends heavily on the map.

Figures 4.3 through 4.6 shows the results of inserting noise randomly into one line of a PNGs using Sub (Figure 4.3), Up(Figure 4.4), Average (Figure 4.5), and Paeth (Figure 4.6) prefiltering methods. Sub's horizontal difference mapping allows for cascades of only horizontally and such only affects the single line modified. If the inserted noise was distributed vertically within the image we would see an effect similar to that of the Up encoding, where we see clear vertical cascades flowing down from the modified pixels. Traversing down the color, the cascades often sharply phases in and out in regard to hue. This is a result of a change in difference causing one or more RGB byte color values for a pixel to overflow. A reversal of this color shift occurs as counterbalancing differences return the cascade back across the overflow. This pattern can be clearly within the eyebrow of Figure 4.4. Dark grays, with a little bit of added color from the

injected noise, transition into a zone of bright colors, which cause red, green, or blue values to overflow generating saturated cyans, blues, greens, reds, purples, and yellows to appear in place of the bright washed out near-white sections in the original.

In the Average encoded PNG we see a much less dramatic response from the injected noise. In this encoding each color-value is stored based on the average of its vertical and horizontal distance. Dramatic shifts in color, seen in Sub and Up methods, don't occur, because any pixel is averaged between two dimensions of change. Color jumps that result of overflow are softened by the averaging operation. Rather than cascades we are smearing, most noticeable in areas that feature subtle or no changes in color originally. The effect of the modified line tapers diagonally, with the encoding's direction of influence. Pixels on the left edge presume a 0-value left neighbor, and such inevitably average away any vertical influence; a pixel's value for any given channel is only half influenced by its upper neighbor, which in turn is only influenced half by its upper neighbor ad nihilum. Therefore after eight or more rows without additional corrupted values, pixels on the left edge return to their initially encoded values. These clean values then slowly average away the corrupted effect left-to-right row by row.

The Paeth encoding, the most complex of the encodings, naturally responds the most chaotically to modification. Unlike the Average encoding, the difference-values that inform Paeth are linked to a single other pixel, (like Sub and Up), but this can be any of three: the pixel to the left, the pixel above, and the pixel on its upper-left diagonal.Three ways for cascades to occur independently across red, green, blue and transparency channels. For any given pixel a corrupted value anywhere in the image above or to the left of that pixel could cascade into it. More opportunities for cascades

means more chances for buffer overflows and radical decoupling of any channel and the value, (brightness), encoded in the original.

The goal behind the pre-filtering process was one of instrumentality: the end is to serve to maximize the effectiveness of the Huffman compression process without losing any data. While at this point, this goal is visible in the outcome, understanding the cascades as a result of this intent, orthogonal to human experience, took experimentation, interpretation, and study of the method. It is this process which defines glitch art practices. This is only one pattern of operation that reverberates out of PNGs when sabotaged.

Another key feature of the PNG file format that is revealed in glitching is its interlacing algorithm. When saved as an interlaced PNG images are decomposed into seven sub-images before being saved sequentially. By transmitting sub-images one at a time a PNG can be displayed client-side at progressively higher resolutions—1/64, 1/32, 1/16, 1/8, 1/4, 1/2, 1—while preserving its overall composition. To generate the sub images (Pass 1 through Pass 7), the initial image data is examined as a series of eight by eight pixel blocks. Pass 1 is constructed from the first pixel of each of these blocks. Pass 2 contains a fifth pixel in each block and doubles the resolution horizontally when added to the first. Pass 3 contains the 33rd and 37th pixel, doubling the resolution vertically when applied. Pass 4 contains the third, seventh, 35th and 37th pixels in each block, doubling resolution again. Each pass alternates interlacing columns and rows until the final Pass 7 adds every other final row (Figure 4.7).

Whether or not a PNG is interlaced is indicated by a single byte in its IDHR (image data header) block. An intervention that changes this single byre reflects across

Figure 4.3: Sub Prefilter Cascade

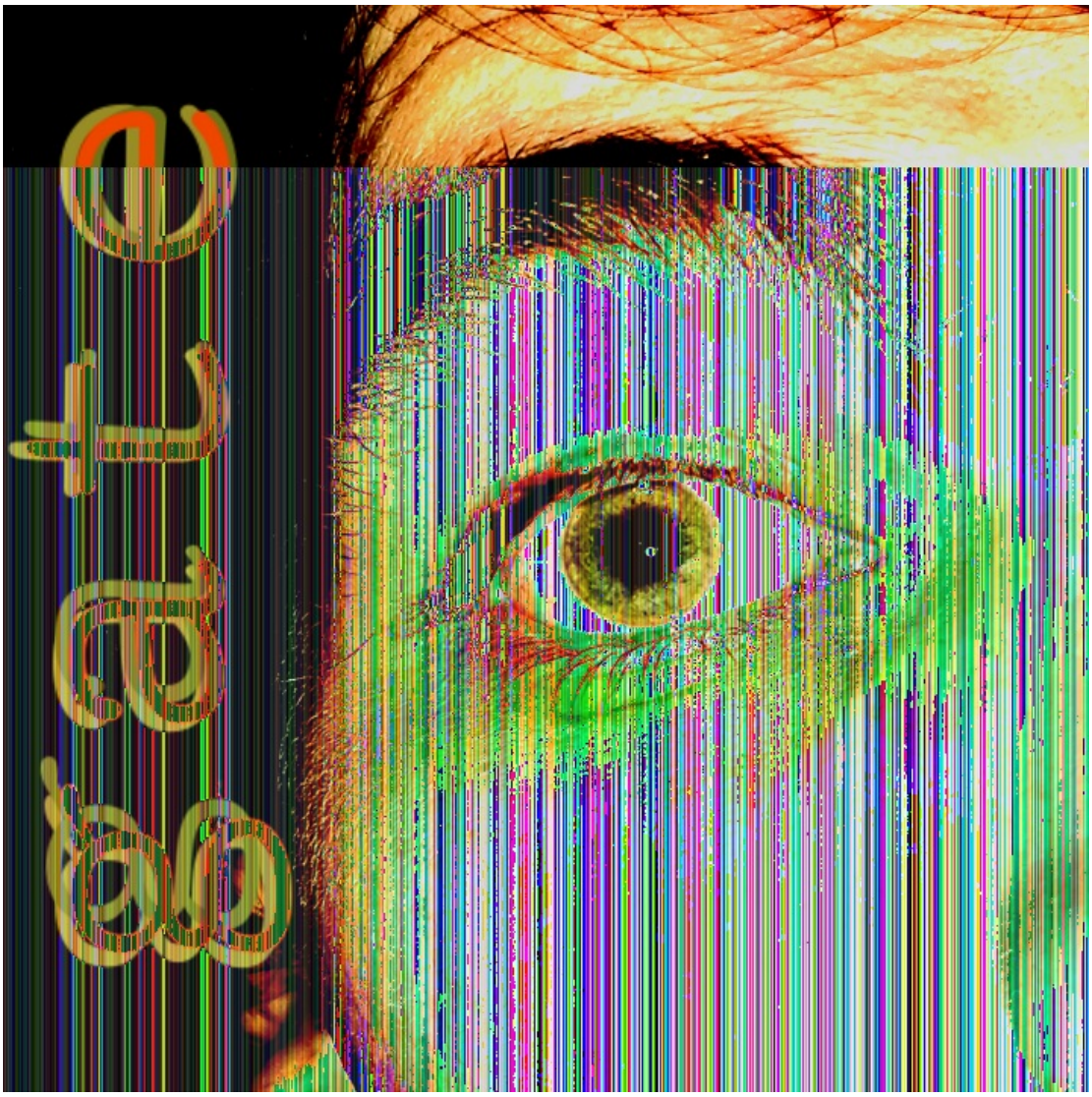


Figure 4.4: Up Prefilter Cascade

Figure 4.5: Average Prefilter Cascade

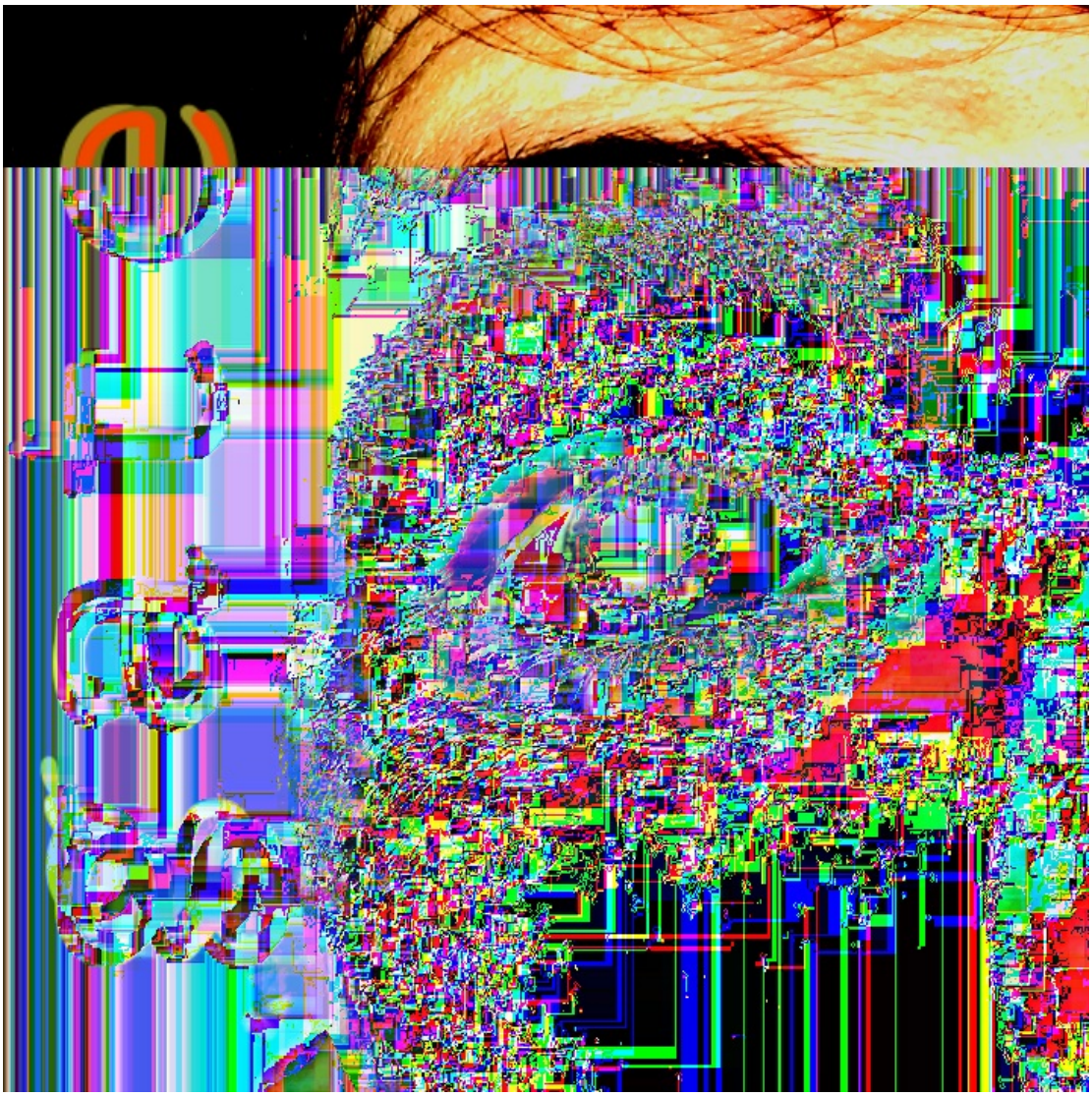
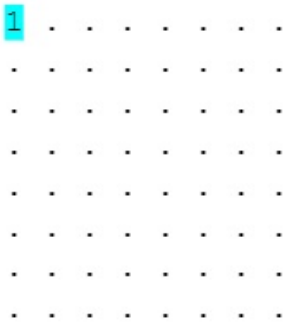
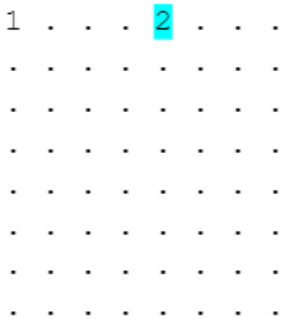


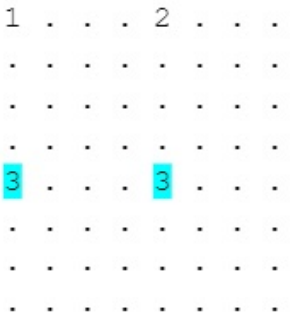
Figure 4.6: Peath Prefilter Cascade



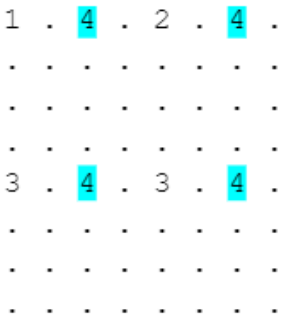
(a) Pass 1



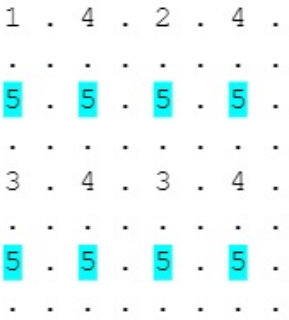
(b) Pass 2



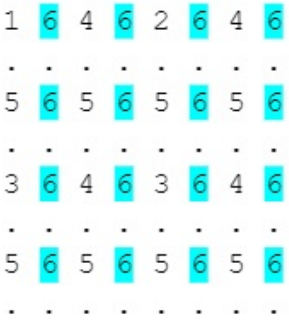
(c) Pass 3



(d) Pass 4



(e) Pass 5



(f) Pass 6

(g) Pass 7

Figure 4.7: Interlacing order of Adam7 algorithm

the entire image. Figure 4.9 shows us a naïve assumption of how these sub-images pixels would be presented when an interlaced image is interpreted as not-interlaced. This is best understood starting from pass seven and working back. Pass 7 starts halfway through the image. It has the same width as the final image, but contains only every other line. It is therefore vertically reduced.

Pass 6 carries a quarter of the overall image, filling the second quarter of the image. However, this pass is compressed vertically as well. Pass 6 may have the same resolution ratio as our starting image, however it is being fit into a container twice as wide, so its first line is rendered and then its second, before moving to the next line. Alternating lines fill up each half-width of the image and the two renderings end up half as tall, again. Pass 5 contains an eighth of the data, but is half as tall relative to Pass 6. It acts like Pass 7 did in the full reference frame, but because it is being split like Pass 6 is as well, it experiences twice the vertical compression. This alternating pattern of half-height leading to more compression to half width leading to reduplication and compression horizontally continues until Pass 1 and Pass 2 which are the same size, each represented by eight hyper-squished cat photos. This is not the whole story though.

Each sub image is ordered as described, but rows in PNG images each contain a filter byte: that tiny number which flags how the rest of the bytes are interpreted. Because interlaced PNG sub-images need to be renderable individually or interlaced with earlier passes, a row-by-row filter pass over the image as a singular whole would pose a problem. Each sub-image therefore has its own set of filter bytes at the beginning of each line. That means, pre-compression, an interlaced image will have more data that a non-interlaced version. This leads to a failed orientation between the image data and

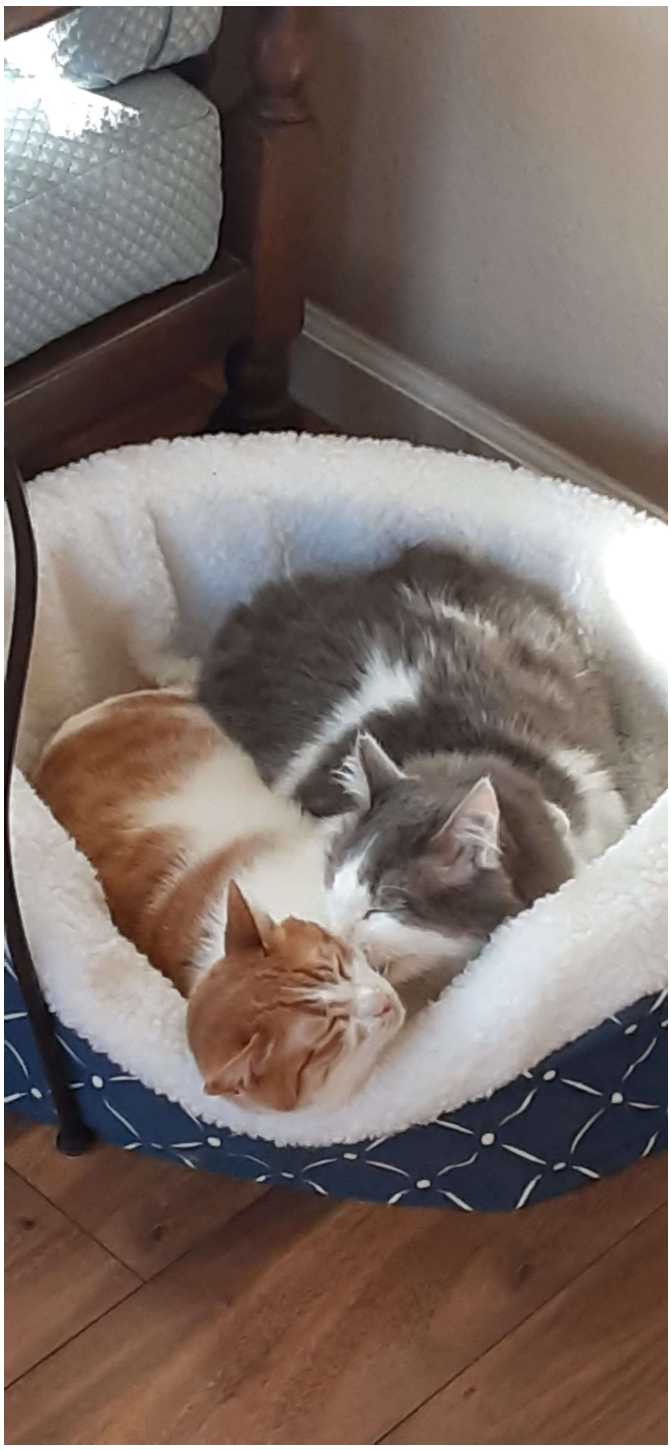


Figure 4.8: Unmodified Interlaced PNG

Figure 4.9: Naïve De-Interlacing Result

the algorithm rendering it. The above example does not take this additional data into account. Because of the additional filter bytes, changing a PNG's interlaced flag leads to images that are the incorrect size and with illegal filter byte values: too broken to be shown in modern browsers; I had to re-export Figure 4.10 via MS Paint to include it here.

So, in addition to recursive patterns we are given an image displayed with mostly out-of-domain value filter bytes defaulting to the None filter. Most programs choose to encode with filters other than None, to optimize compression. The consequence of this is that bytes encoded as difference values are read as color values. Bright colors indicate points where pixels are moving from low values to high values or from one value to a slightly darker value, as a result of overflow. Dark areas where pixels are moving slowly from low values to high values or moving from very high values to very low values are again a consequence of byte-based math and overflow. Middle gray values indicate great differences between adjacent pixels regardless of brightness of the originals.

If we reverse the interlace-byte hack, asking that our PNGs be rendered as if they are interlaced when they are not, we run into the same issues with offset filter bytes. Now there is too little data instead of too much. Turning on the interlacing byte, with no other intervention, like in Figure 4.11, will typically produce the same edge reading effect that we saw in Figure 4.10.

Glitch art is a process of generating new potential through interruptions in computer operations intended to go unseen, which forces the presence of computation-optimized processes into our awareness. The next chapter examines how we can extend

this understanding of a glitch interruption into the domain of narrative production in video games.

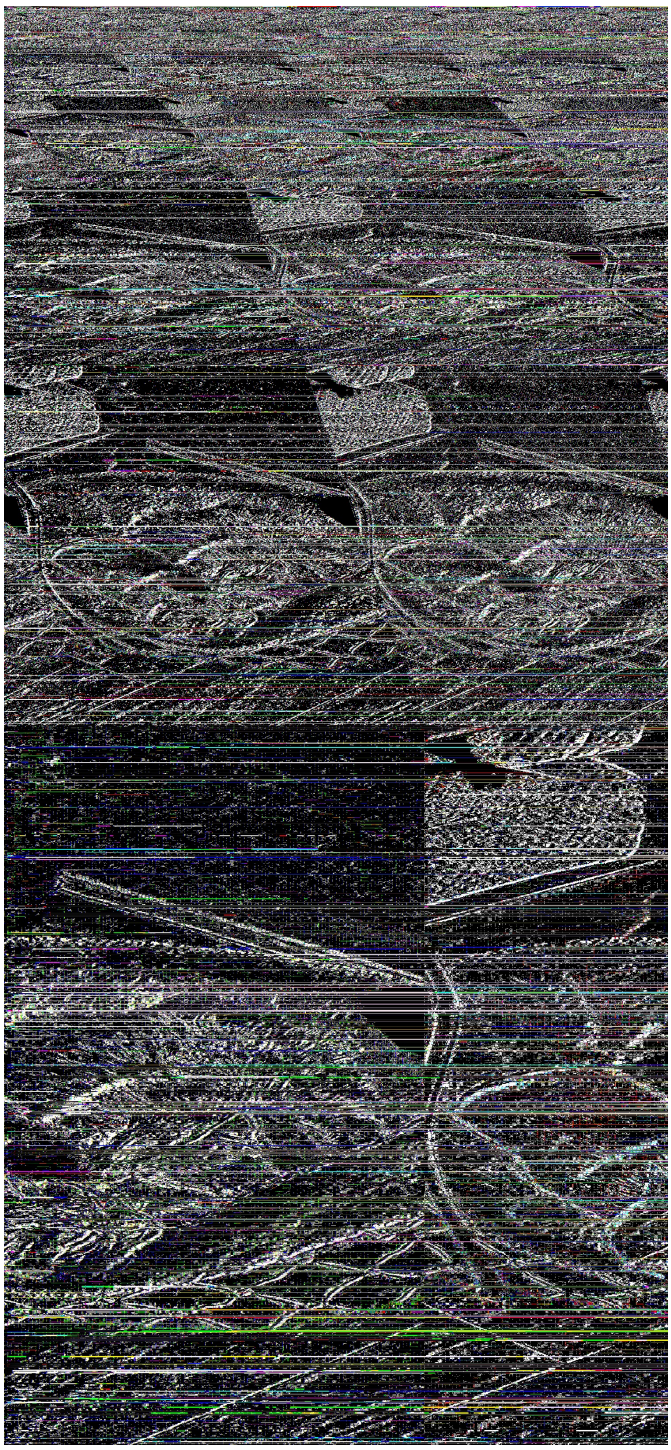


Figure 4.10: De-Interlacing Result

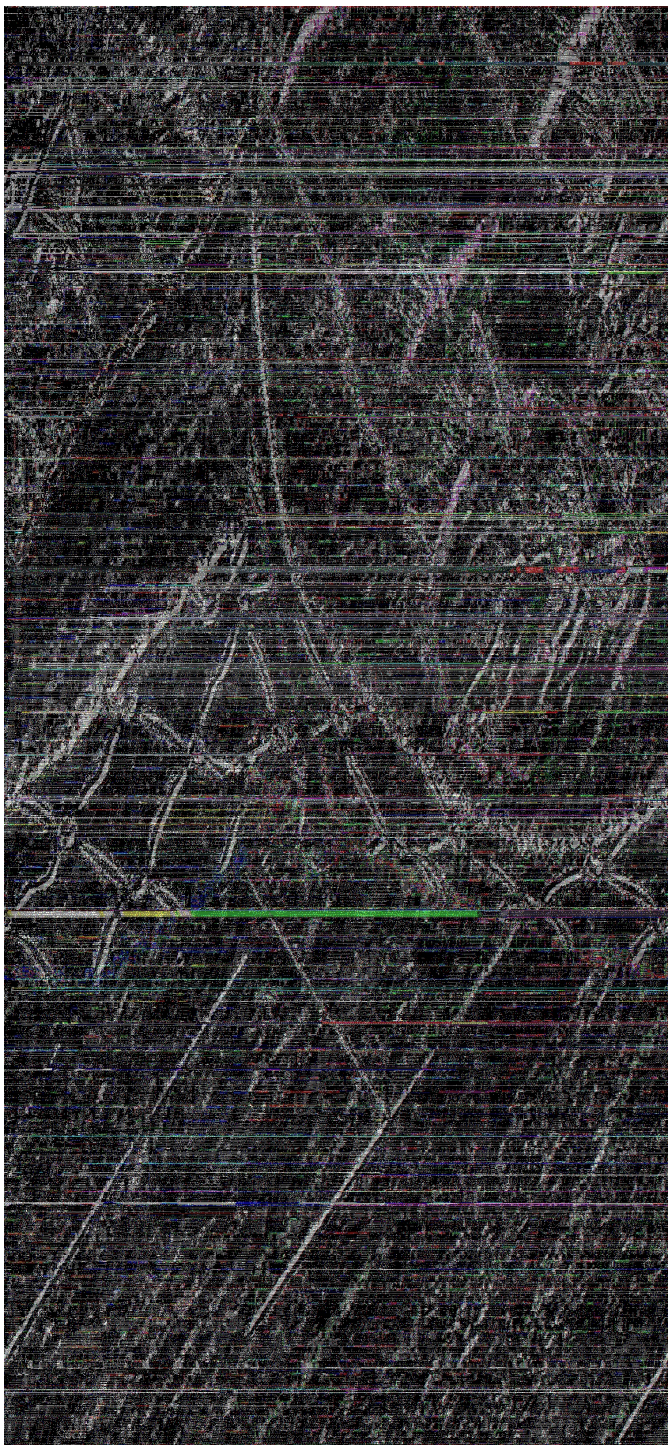


Figure 4.11: Interlacing a non-Interlaced image

Chapter 5

Dramaturgical Technomancy

In our expansion of glitch to breakdowns in cybernetic systems, we must be cautious in regards to the grounds by which we engage in its examination. Glitch has often been used in games for its aesthetic affordances. *Oxenfree*, for instance, uses visual glitches to convey the influence of the uncanny presences which haunt the protagonists [53]. In contrast, this chapter is interested in how it opens up possibilities for emergent narratives. How can we understand glitch as it contributes to transformative modes of play?

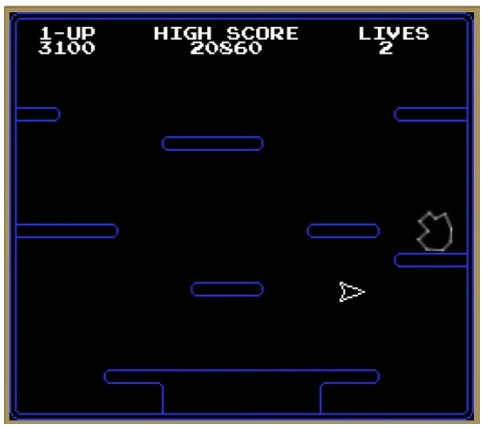
Stefano Gualeni (2019) argues that the "glitch-a-like" behavior provides space to examine how meanings are produced within games.Rather than seeing unexpected glitch as a point of rupture where the materiality of the game resurfaces, Gualeni understands glitches and glitch-alikes signs which recontextualize the spaces they are employed within as a socio-technical phenomenon[23, p. 2]. The glitch-a-like, Gualeni argues, can be used for its potential to unsettle as in the mode of magical realism in literature. However, what Gualeni fails to examine in their analysis is the experiential relation

between a glitch as sign and the logics of play.

Gualeni's key example is the experience of play in *ROM CHECK FAIL!* (2009). *ROM CHECK FAIL!* is contextualized by the practice of games emulation and the surprises which appear while exploring collections of games ROMs which may or may not have been modified. *ROM CHECK FAIL!* remixes the assets, mechanics, and game logics of a collection of classic games: *Super Mario Brothers*, *Space Invaders*, *Asteroids!*, and more. The game is played in levels, each cleared by the player eliminating all of the enemy objects. Every several seconds the game randomizes its visual and musical elements as well as the appearance and behavior of player and enemy sprites. Using the abilities of the characters they are presented with, the player is expected to clear each screen of enemies. In Figure 5.1 the player controls the spaceship from *Defender* and can use its horizontal-only laser to destroy Goombas from *Super Mario Brothers*(a), the spaceship from *Asteroids!* can use its 360-rotation and movement to destroy its classic foe, (b), and Link from *The Legend of Zelda* can approach using his four-directional movement and use his sword on the Crabs from *Space Invaders*, which are trapped in a wedge by their horizontal-only movement (e).

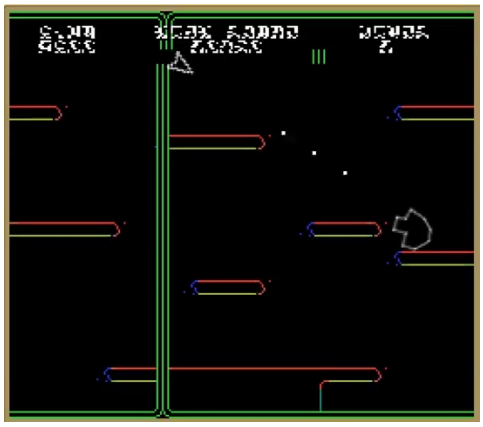
To succeed in ROM CHECK FAIL! the player must develop an intuition for how to best employ a variety of game logics within the game shifting chaos. In Figure 5.1-a, the player prioritizes the enemies that can be targeted most quickly; the Goomba on the right side of the screen is effectively shielded by the terrain so it is left until last. After this task is done the player begins moving towards the right, but a shift occurs transforming their sprite into the *Asteroids!* ship. This form experiences far more inertia and must be oriented based on rotation. This causes the player to overshoot

(a) Time: 2:48



(b) Time: 2:51

(c) Time: 2:53



(d) Time: 2:56

(e) Time: :57

(f) Time: 3:00

Figure 5.1: *ROM CHECK FAIL!* Screenshots[19]

(5.1-c) their movement but also enables them to attack at a diagonal angle not available before (5.1-c). Their attack connects, and the enemy breaks apart into smaller asteroids before the next shift. The smaller asteroids become a stack of crabs. Unable to attack above themselves, the crabs can easily be destroyed by Link.

Situations like this overshooting movement, while temporarily disabling, are an intentional outcome of the game's design. The jank always passes. The core game-loop becomes one of quickly identifying the abilities of the player sprite, the patterns of the enemy sprites, and then employing that knowledge ad hoc and as fast as possible before they change again. There is no access point for misuse in the system for the player.

ROM CHECK FAIL! rewards understanding of the procedural logics of the games it cites and those with an intuition to respond to its protean game space. While it's an ever changing game, the paths to completion are still clear for those familiar with its constitutive mechanics. Any particular combination of mechanics, playfield, and music may be novel, but the game thematically says little beyond that novelty and a seasoned player will not struggle to understand what is possible given any situation based on their literacy of characters and mechanics.

Gualeni's thesis is that the glitch elements, interpreted in and as part of the fiction, provide opportunities for "de-familiarizing and re-ontologizing effects" [23, p. 7]. *ROM CHECK FAIL!* does not serve as a sufficient example of these opportunities. Recombination of images and logics doesn't fundamentally change the goals players are engaging within one step abstracted from the operations of play. The puzzle of correct application of game logics is enabled through the common literacies of its cited games rather than any higher-order break in instrumental play.

Amid the visual and mechanical changes of the game, destruction always remains the game's unifying logic. Glitchiness and shifting relationships becomes something to overcome in this goal rather than the opportunity to produce something new. There is always some type of enemy, there is always some player with a potential means of destroying (projectiles, hopping, consuming, stabbing), and objects which complicate or enable certain movement logics between them. The mechanical glitch-a-like deploys the violent union set of mechanics from its sources, while discarding any of the game ending logics which separate them. The alternate progress mechanics of *Legend of Zelda*, exploration; *Pacman*, consumption; *Super Mario Bros*, traversal, and *Defender*, rescue, are stripped away. Glitch in ROM CHECK FAIL! serves as a meta-narrative for its mixing of historical game intellectual properties and their mechanical relations. The mixing of procedural affordances remains enclosed by a meta-level arcade-completion ontology of play.

Skrillex Quest (2012) performs a similar function when it comes to engaging with glitch. A Flash game, the narrative explicitly takes place within a malfunctioning NES cart, more explicitly anchoring the glitch as a fictional threat and explicit error as an element of an electronic materiality. The conflict is ultimately resolved via appeal to FMV Skrillex as god who must dutifully clear the source of the error by blowing into the game's cartridge. *ROM CHECK FAIL!*'s use of glitch as aesthetic and narrative contrivance is mirrored in *Skrillex Quest* use of glitch to just to justify its throwback to retro material such as the NES The Legend of Zelda, (Figure 5.3). Rather than radically altering or changing the logics of play, the glitch-out enemies function according to the same logics as enemies in any other game, (Figure 5.2-a). Glitch here functions as a

Figure 5.2: *Skrillex Quest* (2012) [1]

narrative sign rather than a disruptive phenomenological event.

As discussed in Chapter 3, games are plays of bodies: systemic objects read as texts through participation and interpretation in the present. Glitch, as an aesthetic of systems exists beyond visual reference and the remix of mechanics within a comprehensible set of possibilities. Rather than understanding glitch as a function in a work that operates on a player I turn to studies of narrative meaning-making in games which center player experience.

Justyna Janik has made an argument for understanding glitch as experience through games through theater studies and phenomenological experience, specifically

(a)

(b)

Figure 5.3: Hidden Reward Caves from *The Legend of Zelda*[56] and *Skrillex Quest*[1]

Tadeusz Kantor's notion of the bio-object. A bio-object is an object produced through the symbiosis of live and artificial elements, such as the sum an actor for its purpose (like a stage), but necessarily provides a purpose for its user. Janik understands glitch as the point of tension in the player-game bio-object where the digital medium itself. When glitches, including interruptions of apparently faulty design, shift games from objects ready-to-hand—invisible and transparent to the player—to the condition of being present-at-hand, the medium of the object is brought into attention, including the agencies outside of the player's control[26, p. 4-5]. An argument that is implicitly aligned with Keogh's frame of games as a cybernetic play of bodies.

Justyna Janik centers how glitches, when noticed, don't break the cybernetic system but rather the players' reading of the fiction. Janik uses the "Suicidal Photographer" from *GTA: San Andreas* and its interpretation by players as of example of this breakage. In the game, one NPC photographs a waterfront and then proceeds to walk into the water and drown. Janik points to the moments of exposure and confusion when a player witnesses this event, uncertain to its purpose and argues that this is the starting point for understanding the medium between the player and game. [26, p. 6]

While Janik's argument is ultimately in line with an understanding of glitch as arising in unexpected breaks in Keogh's play of bodies and engaged beyond surface readings of glitch, it does suggest a greater narrative possibility space where the bio-object conflict can be intentionally invoked and made part of the fiction. As discussed earlier, intentionally designed jank can position the player in tension with the games and its systems. Beyond intentional fourth wall breaks through meta-actions with game consoles draw attention to the medium, such as the Psycho Mantis boss battle from

Metal Gear Solid (1998)—which requires switching which port the controller is inserted into[1]—, fake-out save deletion sequence in *Eternal Darkness* (2002)[48], and even the mandated soft reset in the Sega Genesis *X-Men* (1993) game [9]. All of these are intentional uses, effectively understood through the theatrical bio-object framework and are not misuse or subversive play on the part of the player.

We can pull from fiction and narrative studies to better understand how players can cope with this kind of rupture. Van de Mosselaer and Wildman frame glitches as fictional (mis)communication [52]. Coming from a background in fiction and narrativity, they understand the produced fiction of a work to be the result of a sequence of intended utterances, which are only meaningful when taken together. Unintentional utterances, such as typos in writing or crew members caught on camera in film, can be managed through rejection of them as not part of the fiction. Glitches in games would be considered a form of unintended communication. However, building from Tamer Thabet's understanding that games' narration is produced at least in part by the player [50] they argue that games uniquely provide space for such errors to be made part of the fiction.

Van de Mosselaer and Wildman identify four strategy player's use when coping with the occurrence of glitch and apparent errors within a games system: the weird world strategy, the reconciliation strategy, the rejection strategy, and the disregarding strategy. Both the weird world and reconciliation strategies focus on accepting an element as fictionally true within the text, but vary in dedication to coherence. The application of the weird world acknowledges some element as incoherent to us, but

[1]The designers of *Metal Gear Solid* made sure to include a secondary work around in case the system has a damaged second controller port.

accepts the element as true within the fiction, like Gualeni's magical realist take of *ROM CHECK FAIL!*. Reconciliation attempts to maintain a coherence, through smaller rationalizations of the context. Rejection as a strategy not only includes a player filtering the intentional against the unintentional. It allows the recognition of intentional glitches within a game's narrative, that while representing some aspects of the narrative are themselves non-truths of the fiction. Van de Mosselaer and Wildman use the Scarecrow hallucination sequence in *Batman: Arkham Asylum* to explain this. At one point in the game, the player controlling Batman comes to confront Scarecrow, a villain known for debilitating his opponents using fear gas. At the beginning of the encounter the game seems to glitch and crash. Resuming from the main menu places the player in control of Batman, but now in a surreal context. Fictionally the crashing of the game is not real, but it is metaphorically meaningful for the state of Batman as a focalizer. While this type of narrative play does ask the player to perform a ludonarrative hermeneutic of error, the authorial intention behind the sequence remains.

In "Emergent Narrative and Reparative Play" Grinblat, Manning, and Kreminski apply Eve Sedgewick's reparative reading to the field of ludonarrative hermeneutics—how players go about engaging with the narrative component of interactive digital narratives [22]. They argue players can excavate meaning from interactive digital narratives via two contrasting strategies, reparative readings and paranoid readings, as defined by Sedgewick. Reparative readings are constructed from "weak" networks of "part objects" that together produce meaning suitable to the needs of the reader. Paranoid reading seeks to rediscover the truth or intent within a piece of work. Grinblat et al. extend reparative reading into the realm of narrative sandboxes like *Caves of Qud* or

The Sims, video games with little scaffolding for the player's role in a grand narrative. Production of story in these games is achieved through the active assembling of part objects by the player, in contrast to the uncovering of narrative facts, as seen in investigation games like *Return of the Obra Dinn*. Reparative play builds on games as systems and as texts refusing them only as delivered.

Game glitches are most often not the malfunction of code or hardware, but the unintended consequences of choices regarding game systems and their component entities. When glitches come to the player, they serve as raw material and enable forms of reparative play beyond the imaginings of the original designers, but still of the system those designers constructed. Like in glitch art production, these materials are the result of instrumental processes becoming exposed as a result of unforeseen non-complete failures. *Pokémon Glitch* as both a result of glitch art process and as an interactive game artifact, which breathes as a bio-object according to the strategies brought to it, exemplifies the similarities between glitch art production and reparative play.

Pokémon Glitch is a procedure for producing a playable glitch artifact from a source ROM, designed by Younès Rabii via their modification tool, ROM MUTAGEN [40]. The operation of ROM Mutagen is simple: select a range of data, a source file to mutate, select a likelihood of mutation from 1/1000 to 1, and provide a seed. Each byte is checked and mutated randomly based on these inputs. The tool provides some guidance in producing functional *Pokémon Glitch* ROMs based on Younès' own experiments. Games with a mutation rate of greater than 1/1000 will be likely to crash on boot-up; mutating any of the first 12

Pokémon Glitch is also a game. Like its namesake, *Pokémon Glitch* is a 2D RPG for the Game Boy which combines strategic battles with a monster collection and growing mechanic. Collecting and evolving a variety of the titular Pokémon provides the player with ways to optimize paths through the game's combat challenges. The random changes to source code leads to in-game "part objects" varying wildly from their original form. One of Rabii's examples in their Rouguelike Celebration, from their instance of *Pokémon Glitch* mutated from *Pokémon Gold*, is the diversion of evolution paths from their canonical lines. Their starting fire-type Pokémon, instead of evolving to its final form, evolves into a grass-type creature with two more evolutions to go[39]. The possibility for such a drastic, indeterminate refiguring destabilizes any presumption of the game as a human authored artifact.

Compared to their progenitors, even ludicly-viable instances of *Pokémon Glitch* are unpredictable, hostile, and janky games that force a renegotiation of expectations for players, especially those experienced with the series. Any encounter with an NPC could lead to an immediate reset to the title scene. The glitched nature of the game also allows for one-way passageways where none existed before, leading to situations where saving leads to a soft-lock, demanding a full restart of the game. While the controls are predictable—matching the original game—*Pokémon Glitch* exhibits jankiness through its failure to adhere to the embodied literacies which inform the original. To manage this disconnect, Rabii adapts the literacies of the roguelike genre; interpreting occurrences like the resetting as procedurally generated traps and soft-locking as a permadeath mechanic[39]. While video games, in general, more strongly inscribe their possibilities spaces and presume rejection of breakages fictionally, *Pokémon Glitch*

provides an effective caricature for the possibilities of reparative play.

After experimenting for a few hours and getting many ROMs which crash on boot or perhaps playing glitched out audio loops, or resetting the game on conversations with early NPCs, I produced my own instance of *Pokémon Glitch* by reducing the range of mutation. I came to accept that the game might be less glitchy aesthetically than expected, but still even small transformations led to the generation of interesting formal narrative affordance.

Upon leaving my house I examined the space that is the first town in the game. In the pool of water to the east I noticed an oddity: one of its animation frames is off. There was a halt in the animation every time it cycled. This is a small enough issue to reject and ignore, so I moved on. I collected my first Pokemon, Chikiorita, my childhood favorite, and began on the initial quest to collect a mysterious egg. The few battles with wild Pokémon seemed normal, and I was worried I had been too conservative with the glitch range on the mutagen tool.

I arrived in Cherrygrove City, the second town in the game. I went to the Pokémon Center, the location where players normally go to heal their monsters. Something, however, was certainly wrong. The NPC nurse who heals your pet monsters was not present. I tried to talk towards where she normally is present, but the conversation doesn't trigger. Strategically, this could be worked around. The healing machine in the first town still worked and I could still buy healing items. However, at this point the bio-object of both the game and myself was put into tension. My presumption of the fundamental purpose of the game, as informed by my previous experiences and internalized embodied literacies, was questioned.

I progressed onward and collected the mysterious egg. My following random encounter was a Weedle, a small bug Pokémon, early game fodder. It's strong against my Chikorita because of its poison, but it was still weak. I'm confident. In the first round my monster survived its attack. In the second round the game crashed and reset as the enemies attack animation finishes, the little projectile flew across the screen—it hits. It Killed Me! was my immediate response.

I wouldn't have been so shocked, but the transformation via mutation turning one of the weakest enemies in the game into a run-killer had shaken me. I thought back to the missing medical staff at the Pokémon Center. With only the part-objects provided to me by the game, I came to the conclusion that my human character was poisoned by the attacking Pokémon. If all of the Pokémon medical professionals are missing, then perhaps the human physicians are missing as well. Mutation of random bytes is one form of shaping games before play, but many games directly ask the players to participate in the construction of the part objects which form the narratives they tell.

In this chapter I took a narrative approach to understanding how part object-bodies are repurposed and made meaningful by players in ways outside of an intent presumed in the original artifacts, but video games are never fiction only. Narrative produced through play is only one of many ways that players re-purpose games to produce points of interest and interaction. Speedrunning communities produce categories based through community agreement, dedicated-game streamers often engage in challenge runs which reformulate the spaces of interaction, even playing alone players with deep experience with systems can modulate their experience through process of ad hoc

game design, based on voluntary restrictions. The bio-obect and reparative play can be generalized out of narrative, to all the drama of play.

Chapter 6

Conclusion

Digital computing provides us with a representational working space where specific operations can be performed on data far faster than they can be performed manually. In order to take advantage of this speed, models and protocols must be developed which adapt human methods of understanding the world to this digital space. The primary requirement of doing so is the reduction of representations and working spaces to a collection of logical values and operations between those values. User interfaces are locations where we do work and make meaning. In serving as an abstraction layer they hide lower level computer operations. Through adherence to established protocols, they allow us to take advantage of the accelerated compute space. The price is that we submit to the protocols and models provided to us through the interfaces we use.

Specifications of file types like PNG describe both a set of protocols and the underlying model for what their data is and how they represent it. Programs which follow the protocols of a file type provide the expected output of that type. How-

ever, the output we see is not the data itself. It is an interpretation generated by the protocol before display [11, p. 7-8]. The models that shape this data are not bound by expectations of human comprehensibility and are free to operate instrumentally to minimize secondary costs as long as the expected output is consistent. In the case of the PNG's compression prefilters and interlacing operations work to minimized file size and wait-time. Generally, image manipulation programs minimize the ability to engage with these protocols. In this domain, glitch materiality is a type of productive process that reappropriates the designed protocols that are expected to be hidden. In glitch art production it is experienced in the intentional breach of these protocols to produce results a human being engaging with them as intended would not.

Video games, like any other computerized system, are built upon protocols and models. Their play demands literacies with not just input devices, but also protocols of ludic operation and narrative metaphor. As with any other computerized system, video games feature some protocols intended to produce expected outputs to the players and some which underlie those, making the game models players bring with them, or will be expected to learn, operable by the boolean brain of the computer. Many video games are far more complex systems than most file types, and as I have discussed earlier, the player is expected to become an active component of their operations, but, again, active only according to certain expectations. Here we find glitch materiality at the extremes of play, where players ignore the implied rules of the system and reconstruct it to their ends, whether this means reconfiguring the games' play space through practices of categorical extension[45, p. 69] or in reparative narrative play.

Bibliography

[1] 00beto00. Skrillex quest: 100% playthrough. https://www.youtube.com/watch?v=Dun22eWsy3E, Dec 2012. Accessed: June 11, 2023.

[2] Sara Ahmed. *Queer Phenomenology: Orientations, Objects, Others*. Duke University Press, 2006.

[3] Aubrey Anable. *Playing with feelings: Video games and affect*. U of Minnesota Press, 2018.

[4] Fiona Andreallo. *Mapping selfies and memes as Touch*. Springer Nature, 2022.

[5] Cory Arcangel. On compression. *Noise Not Noise*, 2013.

[6] Kim Arsendorf. Asdfpixelsort. https://github.com/kimasendorf/ASDFPixelSort. Accessed: June 11, 2023.

[7] Andrew Remington Bailey. The museum and the killing jar: How animal crossing's insects reveal videogames' object afterlife. *Loading*, 13(22):7–22, 2020.

[8] Ben Baken-Smith. Glitchbot. http://bitsynthesis.com/glitchbot/. Accessed: June 11, 2023.

[9] Joshua Barsody. X-men: Sega genesis: Reset the computer. http://www.joshuabarsody.com/2013/09/x-men-sega-genesis-reset-computer.html. Accessed: June 11, 2023.

[10] Roland Barthes. The death of the author. In *Readings in the Theory of Religion*, pages 141–145. Routledge, 2016.

[11] Michael Betancourt. *Glitch art in theory and practice: Critical failures and post-digital aesthetics*. Taylor & Francis, 2016.

[12] Michael Betancourt. Glitch art and the cinematic articulation of the 'shot': the convergence of datamoshing with the long take. *Journal of Visual Art Practice*, 21(1):47–71, 2022.

[13] Nick Briz. Thoughts on glitch [art] v2. 0. 2015. *Last Accessed*, 13(10), 2023.

[14] Nik Briz. Glitch codec tutorial, 2011.

[15] Kornhaber Brown. The art of glitch — off book — pbs. https://web.archive.org/web/20120817143755/http://www.youtube.com/watch?v=gr0yi0yvas4&feature=player_embedded, August 2012. Archived: August 12, 2012.

[16] Kim Cascone. The aesthetics of failure:" post-digital" tendencies in contemporary computer music. *Computer music journal*, 24(4):12–18, 2000.

[17] Mia Consalvo. Cheating. In *The Routledge Companion to Video Game Studies*, pages 222–227. Routledge, 2007.

[18] Anne Cranny-Francis. *Technology and touch: The biopolitics of emerging technologies*. Springer, 2013.

[19] Farbs. Rom check fail - 21960 points. `https://www.youtube.com/watch?v=92-1wVZ5-r4`, July 2013. Accessed: June 11, 2023.

[20] Jamie Fenton. Jamie fenton biography. `http://www.fentonia.com/bio/`. Accessed: June, 11 2023.

[21] Matthias Gelbmann and Amanda Delamer. Usage statistics of image file formats for websites. `https://w3techs.com/technologies/overview/image_format`. Accessed: June 11, 2023.

[22] Jason Grinblat, Cat Manning, and Max Kreminski. Emergent narrative and reparative play. In *Interactive Storytelling: 14th International Conference on Interactive Digital Storytelling, ICIDS 2021, Tallinn, Estonia, December 7–10, 2021, Proceedings 14*, pages 208–216. Springer, 2021.

[23] Stefano Gualeni. On the de-familiarizing and re-ontologizing effects of glitches and glitch-alikes. In *DiGRA Conference*, 2019.

[24] Todd L Harper. *The art of war: Fighting games, performativity, and social game play*. Ohio University, 2010.

[25] Paul Hertz. Glitchsort. `http://web.archive.org/web/20210922074924/https://paulhertz.net/factory/2012/08/glitchsort2/`. Archived: September 22, 2021.

[26] Justyna Janik. A game within a bio-object (s): Redefining a moment of the gameplay. In *The Philosophy of Computer Games Conference*, 2016.

[27] Carolyn L Kane. Compression aesthetics: Glitch from the avant-garde to kanye west. *InVisible Culture*, 2014.

[28] Carolyn L Kane. *High-tech trash: glitch, noise, and aesthetic failure*, volume 1. Univ of California Press, 2019.

[29] Brendan Keogh. *A play of bodies: How we perceive videogames*. MIT Press, 2018.

[30] Russell Legacy. *Glitch Feminism: A Manifesto*. Verso Books, London, 2020.

[31] Hugh S Manon and Daniel Temkin. Notes on glitch. *World Picture*, 6:1–15, 2011.

[32] Alan Meades. Why we glitch: process, meaning and pleasure in the discovery, documentation, sharing and use of videogame exploits. *Well Played Journal*, 2(2):79–98, 2013.

[33] Rosa Menkman. Of mimicry and glitch art: The ambivalence of the 'colonial' glitch art discourse. `http://rosa-menkman.blogspot.com/2011/01/monglot.html`. Accessed: June, 11 2023.

[34] Rosa Menkman. A vernacular of file formats. *A Guide to Database Compression Design. Amsterdam*, 2010.

[35] Rosa Menkman. *The glitch moment (um)*, volume 4. Institute of Network Cultures, 2011.

[36] Iman Moradi. Glitch aesthetics. *Unpublished BA dissertation). School of Design Technology, The University of Huddersfield, Huddersfield*, 2004.

[37] Peter Nelson. A game made from other games: Actions and entities in garry's mod. In *The Philosophy of Computer Games Conference*, 2017.

[38] C Thi Nguyen. *Games: Agency as art*. Oxford University Press, USA, 2020.

[39] Younès Rabii. Pokemon glitch - story of a roguelike with no author. `https://www.youtube.com/watch?v=Ove6z8zDsjE`, Oct 2021. Accessed: June 11, 2023.

[40] Younés Rabii. Rom mutagen. `https://pyrofoux.github.io/ROM-MUTAGEN/`. Accessed: June 11, 2023.

[41] Dick Ainsworth Raul Zaritsky, Jamie Faye Fenton. Digital TV Dinner. `https://www.youtube.com/watch?v=Ad9zdlaRvdM`, October 2009.

[42] Greg Roelofs. Uportable network graphics (png) specification and extensions. `http://libpng.org/pub/png/spec/`. Accessed: June 11, 2023; Last modified 29 September 2011.

[43] Greg Roelofs. `http://www.libpng.org/pub/png/pngpic2.html`. `http://www.libpng.org/pub/png/pngpic2.html`. Last modified 25 April 2021.

[44] Nadja Sayej. Glitchr is the most interesting artist-hacker on facebook. `https://www.vice.com/en/article/mgpv9a/glitchr-is-the-most-interesting-artist-hacker-on-facebook`, March 2014. Accessed: June 11, 2023.

[45] Madison Schmalzer et al. Transition games: Speedrunning gender. *Dissertation*, 2022.

[46] Madison D Schmalzer. Janky controls and embodied play: disrupting the cybernetic gameplay circuit. *Game Studies*, 20(3), 2020.

[47] Ant Scott. Glitch art blog. `http://www.beflix.com/archive/blog0701.html`, Jult 2001. Accessed: June 11, 2023.

[48] SuperHobbit. Eternal darkness - all sanity effects. `https://www.youtube.com/watch?v=RSXcajQnasc`, November 2011. Accessed: June 11, 2023.

[49] Jan Švelch. Negotiating a glitch: Identifying and using glitches in video games with microtransactions. In *New Perspectives in Game Studies: Proceedings of the Central and Eastern European Game Studies Conference Brno 2014*, pages 55–70. Masarykova univerzita, 2015.

[50] Tamer Thabet. *Video game narrative and criticism: Playing the story*. Springer, 2015.

[51] Adam Trionfo. Programmers of the bally arcade/astrocade built-in programs august 17, 2016. `https://ballyalley.com/faqs/Programmers{%}20of{%}20the{%}20Astrocade{%}20Built-In{%}20Programs.txt`. Accessed: June, 11 2023.

[52] Nele Van de Mosselaer and Nathan Wildman. Glitches as fictional (mis) communication. *Miscommunications: Errors, mistakes, media*, pages 300–315, 2021.

[53] Agata Waszkiewicz. Glitch as the representation of the uncanny in oxenfree (2016). *Homo Ludens*, 12(1):213–225, 2019.

[54] Michele White. The aesthetic of failure: Net art gone wrong. *Angelaki: Journal of Theoretical Humanities*, 7(1):173–194, 2002.

[55] Dan Zahavi. *Phenomenology: the basics*. Routledge, 2018.

[56] The legend of zelda walkthrough – the gathering. `https://www.zeldadungeon.net/the-legend-of-zelda-walkthrough/the-gathering/`. Accessed: June 11, 2023.