Towards Automatically Abridging Game Levels

Varun Bhat[1]**, Gillian Smith**[1]
[1]Worcester Polytechnic Institute
100 Institute Road
Worcester, Massachusetts 01609

Abstract

In this paper, we describe an approach to procedurally generating abridged game levels. We present a design pattern-based method for abridging a collection of Super Mario Bros. levels. These design patterns are salient regions that users describe as memorable after playing original Mario levels. We use the Mario AI Framework and Mawhorter and Mateas's Occupancy-Regulated Extension (ORE) generator to generate 20 abridged levels based upon 3 original levels from Super Mario Bros. We conduct a preliminary study for 2 abridged levels and compare the play experience with the original levels' play experience. We also compare these abridged levels to those created by previous level generators using the same patterns. Finally, we offer a critical reflection of the process taken to generate these levels, and the assumptions and values inherent in this design pattern-based approach.

Introduction

Most forms of media today have a tradition of abridgment. Works of literature may be shortened to account for different audiences who have different cognitive abilities and/or time to devote to the original work [1]; theatrical productions are often shortened for film adaptations; films are edited for time and content in order to be shown on short plane journeys on small screens; songs may be edited to fit radio formats [2] Games, too, often are subject to abridgment, most often through written reviews, gameplay trailers, or *Let's Play* videos. However, these methods do little to provide players with the experience of gameplay. Demo levels, though playable, typically focus on early gameplay rather than offering an abridgement of the full game. Previous works of creating abridged video game content have resulted in the creation of narrative summaries, and video montages (Barot et al. 2017; Cheong and Young 2006; Mindek et al. 2015; Cheong et al. 2008). Abridgment in games is underexplored, however; some games do allow either narrative or combat elements to be skipped or auto-played. For example, the "Assist mode" in *Super Mario Odyssey*(Nintendo 2017) or skipping dialogues, and cutscenes in XCOM 2 (Firaxis Games 2016). We are not aware of any editions of games that have been deliberately abridged.

Our main motivation for this paper and work was to create an abridgement of a game that portrays the highlights of the full, original game. This could give a player a quicker way to enjoy the salient parts of the original game without spending too much time in it. It would especially be useful for people who don't have the time required to explore the full game.

In this paper, we present a method for procedurally generating abridged levels based on a collection of Super Mario Bros.(NES, 1985) (Shigeru et al. 1985) levels, using the Mario AI Framework 10th Anniversary Edition (Khalifa 2019). We choose to abridge the game with respect to level design elements that players consider *memorable*; we operationalize "memorability" as user-identified regions of a level that players claim to be significant after having played. We chose "memorability" because an initial goal with our work was to generate a preview level that lets a player decide if they are interested to play the full game, akin to a trailer for a movie. Thus, the abridgement would more coverage of the game than a demo level.

We performed a user study to identify level regions that players remarked upon as memorable and important. Based on user-identified regions, we then created a saliency map for each level and extracted high saliency regions as design patterns. We then used the Occupancy-Regulated Extension (ORE) generator (Mawhorter and Mateas 2010) to assemble these design patterns into abridged levels. We focus specifically on level design (to include level structure and art assets), acknowledging that games incorporate so many different and interlocking creative elements (Hendrikx et al. 2013; Liapis, Yannakakis, and Togelius 2014), and that aiming to procedurally abridge all is outside the scope of this early research.

We begin this paper by introducing how both abridgement and summarization has been explored in several forms of media including video games. We then give a brief overview of how design patterns are used in level generation, and how they are derived. We describe the methods used for salient

[1]As an extreme example: the *Cozy Classics* book series (Wang and Wang 2012) takes works of literature such as Jane Austen's *Pride and Prejudice* and abridges them to twelve words, each illustrated with needle felted scenes, for preschool children to enjoy.

[2]Perhaps *too* extreme, though nonetheless relevant: a Canadian radio station experimented with halving the length of songs they played to fit more music per hour (Jones 2014).

region identification, and describe how we selected salient features and incorporated them into level generation. We conduct a comparative study between 2 generated abridged levels and the original Super Mario Bros. levels. This study compares adjectives of play and asks participants to judge the better abrdiged level. Finally, we close with a critical reflection on our approach to this research, on the underlying values and assumptions in design pattern-based generation and what that means for applications in summarization and abridgment, and how we think this work can be improved in the future.

Related Work

An abridgement is generally shortening a pre-existing authored work to include elements that the editor considers to be "highlights" worthy of inclusion. Automated summarization and abridgment systems already exist for a variety of media including text, audio, and video using methods such as tokenization (Hovy and Lin 1998), graphs based on lexical analysis (Liu et al. 2017; Erkan and Radev 2004), Hidden Markov Models (Conroy and O'leary 2001), video analysis and segmentation (Lienhart, Pfeiffer, and Effelsberg 1997), and audio signal analysis and processing (Peeters, La Burthe, and Rodet 2002; Spina et al. 2017). Summarization in video games is complicated, complex, and deep. As argued by (Hendrikx et al. 2013) and (Liapis, Yannakakis, and Togelius 2014), a video game consists of many different and interlocking creative components. The complexity of a video game and the broad variety of content that it includes poses a significant challenge in creating a summary that essentially captures all that the game is trying to convey. A summary can also vary according to what a user requests (Hahn & Mani, 2000). For example, a summary of character biographies in a video game would be different than the summary of its plot.

Prior work in summarization for video games focuses on converting gameplay into another format, such as a video trailer or written description of play. Mindek et al. proposed a method of summarizing gameplay highlights into a video montage by applying stop-motion capture to camera views of players in-game (2015). The "Steam Microtrailers" from Steam Labs (SteamLabs 2019) also generates abridged video trailers from video input, sourced through existing promotional material provided by game developers. Originally created by Ichiro Lambe as a Twitter bot (Lambe 2016), these generated trailers composite 6 seconds of snippets from available videos of the game to create a single, smaller 'micro trailer' of the game. Panagiotopoulos, Giannakopoulos, & Liapis provide a method for summarizing multiple text-based game reviews, using keyword and sentiment analysis to determine which features should be included in the summary (2019).

In contrast with prior work on game summarization, we focus on creating *playable* abridged levels, operating directly on level structure data as both input and output. Rather than inferring salient features from video or text (though, in the case of Mindek et al.'s study, their input video comes from human players), or drawing from already-curated video inputs, we determine salient features via an empirical study of how users interact with and react to original levels. Through this, we aim to come closer to understanding how level design factors drive player experience of the game.

Overview of Abridgement Process

1. **Phase 1: Identifying "Memorable" Level Regions.** We study how users play levels and interview them to determine which areas of the levels they find the most memorable and mark it on a printed map.
2. **Phase 2: Distilling Salient Design Patterns.** Using data from phase 1, we determine salient regions and identify the most salient as design patterns. The salient regions of all playtesters are combined into a single salience map per level. Then, we use quantitative analysis on the maps in order to extract individual salience features or 'chunks'.
3. **Phase 3: Generating the Abridged Levels.** This phase deals with using the extracted chunks and combining them as input to the ORE generator in order to generate our abridged levels. It also deals with testing these levels and comparing the experience with the original game experience. We do this in order to gauge how close they were to abridging them through another user study.

We ran studies and generated levels using the 10th Anniversary Edition of the Mario AI Framework. It's a research-based framework with an in-built, playable emulator, making it a suitable testbed for this project. We modified the framework to play the original Mario audio during Phase 1 playtesting, and to support the Mario respawning in the level if he loses a life.

Identifying "Memorable" Level Regions

To identify regions that are "memorable" and thus worth including in our abridged levels, we first conducted a user study where the participants played 3 levels from the original Super Mario Bros. and reported upon sections that were particularly important or memorable to them.

Level Selection The levels we chose for this study are 1-1, 2-1, and 4-1 [3]. Since our method of recruiting participants did not screen for prior experience with the game, we wanted to choose levels appropriate for less experienced players. However, we also wanted sufficient level variety to introduce new elements of the game for the abridgment process. Thus, we selected introductory levels from different worlds. We chose worlds that share the same visual style (thus, World 3 is excluded) because the generator does not support multiple tilesets, and we wanted to maintain visual uniformity for this first attempt at abridgement. We didn't include the overground and the underground levels for similar reasons.

We also developed a custom tutorial level to orient players to the game. Since the original game is a controller-based but participants played with keyboard and mouse, we wanted to provide maximum familiarity to both experienced and new players to make their experience smoother. The tutorial

[3] Super Mario Bros. is divided into eight worlds, and each world has four levels (Wiki 2020).

level was custom-designed and contained 1 gap, 1 enemy, 1 powerup, and 1 obstacle. We use this to familiarize them with the mechanics and level elements of the larger levels.

Method We asked participants to play the 3 levels, and recorded both keyboard inputs and vocal expressions during play (via audio recording). Users played through the tutorial level, and then played each of the three original game levels until a maximum of 15 minutes per level or till satisfaction. The time limit was chosen in order to limit the total study time to no more than 1 hour per participant. The players were encouraged to play naturally, and were told they could vocalize while playing if they wished to do so. The goal was to provide as much freedom as possible to simulate a comfortable gameplay environment.

During playtesting, one researcher sat with the participants and observed their play. The researcher held three level maps, one for each level which were obtained from the Video Game Level Corpus (VGLC) (Summerville et al. 2016). The researcher marked areas or elements that the participants remarked upon, as potentially salient along with noting down their expressions and/or words.

Immediately following their play experience, the players were asked to circle or outline sections on their own printed map that they felt were memorable. They were given the option to watch a video replay of their gameplay. They were also asked to write down any specific emotions or thoughts that they felt were important to be linked to these sections.

While the players marked these sections, a short, semi-formal interview was conducted asking the players some questions regarding their gameplay. The questions were aimed towards understanding what key objects played a pivotal role in influencing the player's gameplay experience. The questions were as follows:

1. Which parts of the levels did you find most memorable?
2. Which characters did you like in the gameplay?
3. Which was your favorite game object to interact with?
4. On a scale of 1-5, how satisfied are you with your gameplay?

Finally, we asked participants to complete a survey that asked for their age bracket, their preferred genre of video games, their frequency of play, and prior Super Mario Bros. (NES, 1985) experience.

Gender and race were intentionally left as free-entry, in keeping with current best practices for inclusive survey design with a small number of anticipated responses. (Spiel, Haimson, and Lottridge 2019)

Participants We recruited participants by advertising via email to department mailing lists.

18 participants completed this stage of the user study, all aged over 18. Out of the 18 participants, 17 were aged between 18-28 while 1 participant was between 51-61. 9 participants (50%) reported that they enjoyed playing platformers. The most popular category was the Action genre at 14 participants (78%). 10 participants (56%) reported playing games Daily, 4 Weekly (22%), 2 Monthly (11%), 1 Never (5%).

13 out of the 18 participants (72%) had experience in the Super Mario Bros. (NES, 1985) while 4 participants (22%) had experience with the Super Mario universe on other handheld devices (e.g. Wii, Original Gameboy, SNES). 1 participant chose not to answer this question.

All 18 participants self-reported gender and race using their own descriptors. 7 people (39%) identified themselves as 'Female', 10 people (56%) identified themselves as 'Male', and one chose not to respond. 12 people (67%) identified themselves as 'White'. one (5%) as 'half Chinese, half White', one (5%) 'African American', one (5%) 'Indian', two (11%) 'Asian', and one chose not to respond.

Salience Maps

After the playtesting sessions, we transcribed each annotated level map (user-provided and researcher-annotated) as a 2D array, where each array element corresponds to a 1x1 tile on the original map. We assigned all regions marked as salient on the map a value of '1', and '0' otherwise. Several annotations included partial tiles; caused if a user circled a region where the boundary tiles were partially marked. These partially annotated tiles were marked as salient only if they contained a game object (i.e. not open space). We summed together all annotated maps for a given level to create a frequency-based saliency map for each level. Figure 1 shows the 1-1 level map with the saliency map overlaid. Note that highly salient regions emerge from multiple users annotating them, and variability in how users marked their maps results in a frequency fall-off surrounding salient regions. We created saliency maps for all three levels.

Distilling Salient Design Patterns

From these saliency maps, we extract slices that can be used by the ORE generator. The maximum frequency in each saliency map is f_{max}. A slice is a set of contiguous tiles in the saliency map whose frequencies are within X units of f_{max}. Figure 2 shows two different slices with different X values: the first is contiguous tiles that have the value f_{max} ($X = 0$), the second is contiguous tiles whose frequencies fall between $[f_{max} - 1, f_{max}]$ ($X = 1$). As mentioned in describing the creation of saliency maps, there is a fall-off in frequency surrounding salient regions due to difference in how users annotate their maps. For this study, we manually selected a X threshold that we determined by visually inspecting potential libraries resulting from different X values, and choosing a threshold that minimized repetition between patterns while maintaining surrounding context. In future work, we aim to find a more consistent method for determining the X threshold.

As a result of this process, we have a library of level slices for use by the ORE generator. From Level 1-1 there are 56 slices (X = 13, $f_{max} = 17$), from Level 2-1 there are 100 slices (X = 20, $f_{max} = 23$), and from Level 4-1 there are 68 slices (X = 10, $f_{max} = 12$).

Generating Abridged Levels

We use the ORE generator to combine these level slices from multiple levels into a set of potential abridged levels.

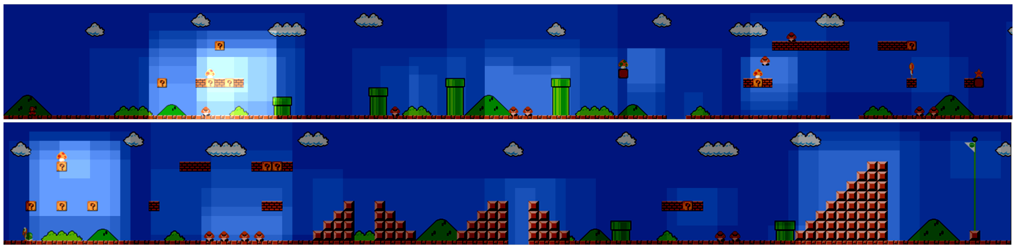

Figure 1: Contrast map of salient regions in 1-1 (Split into two halves for better readability)

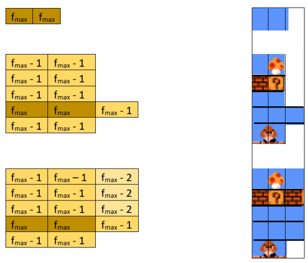

Figure 2: Slices vs. their respective level slice

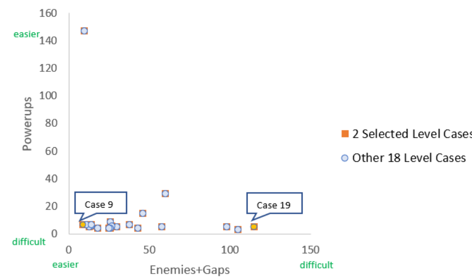

Figure 3: Scatter plot showing how generated candidate levels vary in terms of number of enemies+gaps and powerups.

Why ORE?

The ORE generator recombines pre-authored, arbitrarily sized "chunks" that are annotated with 'anchor' points that mark how they can connect to other chunks. It allows for weighting the selection of chunks according to desired frequencies. This makes it a good match to the input data we have available: arbitrarily-sized salient design patterns with associated frequency data. We manually add anchor points to our level slices.

Placing Anchor Points After converting our level slices to ORE's ASCII format, we manually selected locations for anchor points in each slice according to the following two rules: 1) anchor points must be placed on a platform in order to eliminate the possibility that Mario starts in mid-air; and 2) anchor points cannot be immediately adjacent to enemies, in order to give Mario a safe starting point.

Weighting Slices The ORE generator uses the frequency values identified for each slice (as shown in Figure 2), which is then normalized to the domain of [0,1] for the weighing process of generation.

Resulting Levels

The ORE generator can create many different potential abridged levels from the library of salient patterns we provide. We generated 20 candidate levels. The ORE generator does not make strong guarantees about the validity of generated levels (i.e. ORE can generate levels that are impossible to play even for skilled players), and weighted-random assemblage of chunks can lead to emergent designs that are extremely difficult. From the set of candidate levels, we first excluded levels that met the following criteria in relation to overall level playability: 1) We should avoid have recurring gaps between single ground tiles, as the player must slow down considerably to time precision jumping, and 2) We should avoid having several enemies clustered around a single area such that it is challenging to navigate.

We then chose two final abridged levels out of the remaining candidate levels. To aid in selecting these two levels, we estimated the difficulty of each level by plotting the number of enemies and gaps against the number of powerups. Figure 3 shows how levels distributed across these metrics. Figures 4 and 5 show two abridged levels that occupy different extremes of our metrics.

Evaluating the Two Abridged Levels

Method Due to COVID-19, our study took place over Zoom and Discord meetings in a one-on-one setting. Participants were sent all required files to run the study on their own computer. The study had 2 phases. In the first phase, we asked the participants to play the 3 original levels used in developing the pattern library i.e., 1-1, 2-1, and 4-1. This was to help them get familiarized with the source content. The players were encouraged to play naturally, and were allowed vocalize while playing if they wished to do so.

Immediately following their play experience, the players were asked to fill out a survey. The questions in this survey were aimed at understanding the participant's play experience by having them identify, choose, and report adjectives that described their play experience (Thominet 2016; Miaoqi Zhu and Moser 2017):

1. Please choose the adjectives that describe your "entire" play experience: (Each participant could select multiple

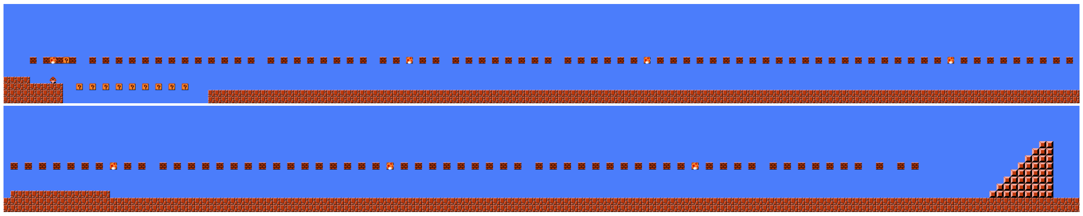

Figure 4: First case of the two abridged levels (Split into two halves for better readability)

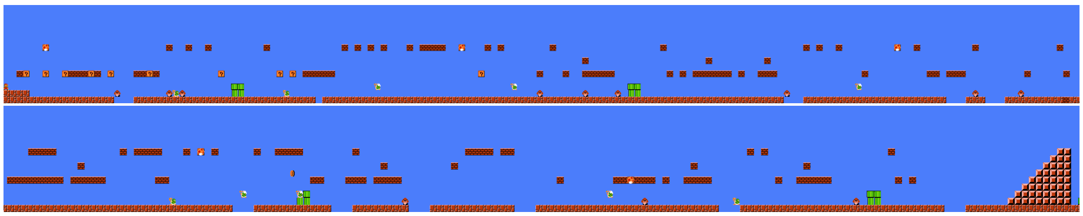

Figure 5: Second case of the two abridged levels (Split into two halves for better readability)

adjectives for Question 1)- *okay, meh, enthralling, interesting, simple, lacking, boring, drab, difficult, rich*

2. Please type in any additional adjectives that may apply to you.
3. Please type in 5 moments that you thought were"memorable" in your play experience

The listed adjectives were chosen for their complicated and abstract emotion that they each represented. For example, a play experience could be "frustrating", and yet be "interesting". We also chose opposing adjectives and mixed them in with emotions that signified a more neutral experience. We also allowed the participants to self-report adjectives in the next question. Instead of collecting level map data, we ask participants to tell us the memorable moments through a written description. We asked the same demographic and game experience questions as in our prior study.

Then, we asked our participants to play through our two abridged levels under the same conditions as the previously described play experience. Immediately after their gameplay session, the participants were given another survey that asked them to identify adjectives that described their abridged levels' play experience from the same list. Furthermore, we asked them to tell us which of the two abridged levels was better, and why.

Participants We recruited participants by advertising via email to department mailing lists as well as through online Game Development communities. 5 participants completed this stage of the user study. All participants belonged to the age group of 18-28. 2 participants reported that they enjoyed playing Platformers. The most popular categories were Action and Role-Playing Games. 1 participant reported playing games Daily, 3 Weekly, and 1 Monthly.

1 participant had experience in the Super Mario Bros. (NES, 1985) while 3 participants had experience with different Mario titles on other handheld devices (e.g. Wii, DS). 1 participant had only seen Super Mario but had never played.

All 5 participants self-reported gender and race using their own descriptors. 1 participant identified as "Pangender", 2 as "Female", and 2 "Male". 1 participant identified as "Irish-Mexican", 1 as "Caucasian", and 3 as "White".

Results All 5 participants reported the same abridged level (codified as Case 19 (Figure 5) in our data) to be the better abridgment. Table 1 shows a comparison of adjectives chosen by each participant in describing their play experience. We observe that participants tended to identify the same or similar sets of adjectives to describe the original levels' play experience as well as the abridged levels' play experience.

We also observed that in most of the identified memorable moments, participants seemed to find sections/areas with a certain level of mechanical complexity to be worth mentioning. This is either characterized by avoiding or killing enemies, jumping over them, grabbing powerups, and being able to clear levels. These kinds of memorable moments appear more frequently in Case 19.

Discussion

Our attempt at automatically abridging levels has revealed many more questions than answers. Here, we offer a reflection upon what we have learned from the process of trying to procedurally abridge a game, including how design pattern-based generation fits this problem space, and what new research opportunities arise from it.

Pattern-Based Abridgement For the purposes of this project, we defined an abridged level as a collection of patterns from levels that users describe as memorable. Through our adoption of the ORE generator, we further make the assumption that the ordering of these patterns does not matter, and that the probability of their appearance does. To our knowledge, there is no off-the-shelf, pattern-based generator for Mario available that takes into account pattern ordering or gives any meaningful control over it. Though there has been work in progression-based level generation for plat-

ID	Original	Abridged
1	okay, frustrating, because lag	okay
2	okay, meh, enthralling, frustrating	okay, meh, simple, indifferent
3	interesting, difficult	difficult, Artificial, odd
4	interesting, simple, difficult	enthralling, interesting, simple, difficult, mysterious (cause the levels were a little weird), unnerving (for the same reason)
5	okay, interesting, focused, frustrated, satisfied	interesting, difficult, engaged, frustrated

Table 1: Adjectives used by each participant to describe original and abridged play experiences.

formers (Green et al. 2018), the system uses an agent-based evolutionary approach and does not allow for tight, authorial control over chunks or progressions.

User-Defined Saliency Patterns Our method for gathering input patterns to ORE relies upon user definition of salient regions. However, users choose "memorable" regions for different reasons. For example, one playtester notes the importance of being challenged:

> "lots of enemies, thought I could jump over, neat challenge"

Another playtester has more nostalgic reasons and says:

> "happy memories, subtle tutorial"

Some playtesters state reasons related to the design choices in the level, such as one user who remarked:

> "easy and fun, extremely difficult flower to grab, multiple routes"

Another playtester referred to a previous level's mechanics:

> "open field [in 4-1] led to a faster-paced level compared to the multi-sectioned 2-1"

These kinds of differences in motivation for choosing memorable patterns–level difficulty, nostalgia, risk/reward, and comparison to other levels–cannot currently be accounted for in our generation process. Interesting future work may be to incorporate tag-based systems into generators, and permit selection of level elements from common sets of tags. However, even then, each users' choice of what is memorable is inherently personal, and it is difficult to know how to treat this in creating a level abridgment (whether automated or not). By treating each users' selections as equally important in generating saliency maps, we lose these personal stories. Further interesting future research may be in looking at how individual users can create abridged levels for others to play (sharing what was important about their experience with a friend, for example), thus highlighting their own ideas for what was memorable/important, rather than attempting to create a more "general" abridged level.

Variability in Generation The approach we took to identifying salient patterns in the input levels resulted in hundreds of design patterns as input to the ORE generator: far more than could be incorporated into any single abridged level. As a result, there is high variability between even valid and playable generated levels. In future work, we intend to investigate methods for identifying salient regions of levels that permit user flexibility in identification, respect all users' opinions, yet do not result in repetitive patterns.

Using Super Mario We explored procedural abridgement using Super Mario Bros. as a testbed due to the depth of prior work on this game and availability of generators and a framework (Horn et al. 2014). Furthermore, the levels in Super Mario Bros. are modular and could be played independently of each other. However, we recognize that our approach to identifying memorable regions, mapping saliency, and generating summary levels may not apply well to other games. Platformer games that have different movement mechanics and camera perspectives (e.g. vertical movement, or 3D movement) may have more complex design patterns and pattern re-assembly requirements. Further, games that have complex narrative or logics other than spatial movement may also require abridgement approaches that don't lean as heavily on spatial design patterns.

Future Work

This work can be expanded upon in several directions. The foremost concern would be to expand the generative space to greater than 20 abridged levels. This would provide a greater variety and options in terms of level curation and selection. It would also be interesting to see if salient features of a level can automatically be inferred through bot playtraces, or previously identified level features. The existing work could also have a larger participant pool. Finally, more study is needed to judge how this approach could be applied to different games and/or genres(within the platformer genre and outside), and different methods for identifying appropriate design patterns to include in summary levels.

Conclusion

In this paper, we presented a method for extracting salient regions from existing game levels, converted them into design patterns for use in a pattern-based level generator, and used these to generate single-level abridgements of a game. This exploration into generating playable, abridged levels has improved understanding of the strengths and weaknesses of the ORE generator through its application, and uncovered new avenues for research in procedural content generation.

References

Barot, C.; Branon, M.; Cardona-Rivera, R. E.; Eger, M.; Glatz, M.; Green, N.; Mattice, J.; Potts, C. M.; Robertson, J.; Shukonobe, M.; et al. 2017. Bardic: Generating multimedia narratives for game logs. In *Thirteenth Artificial Intelligence and Interactive Digital Entertainment Conference*.

Cheong, Y.-G., and Young, R. M. 2006. A framework for summarizing game experiences as narratives. In *AIIDE*, 106–108.

Cheong, Y.-G.; Jhala, A.; Bae, B.-C.; and Young, R. M. 2008. Automatically generating summary visualizations from game logs. In *AIIDE*.

Conroy, J. M., and O'leary, D. P. 2001. Text summarization via hidden markov models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 406–407.

Erkan, G., and Radev, D. R. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research* 22:457–479.

Firaxis Games, F. I. 2016. XCOM 2.

Green, M. C.; Khalifa, A.; Barros, G. A. B.; Nealen, A.; and Togelius, J. 2018. Generating levels that teach mechanics. In *Proceedings of the 13th International Conference on the Foundations of Digital Games*, FDG '18. New York, NY, USA: Association for Computing Machinery.

Hendrikx, M.; Meijer, S.; Van Der Velden, J.; and Iosup, A. 2013. Procedural content generation for games: A survey. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 9(1):1–22.

Horn, B.; Shaker, N.; Smith, G.; Togelius, J.; and Dahlskog, S. 2014. A comparative evaluation of procedural level generators in the mario ai framework. In *Foundations of Digital Games 2014*. Society for the Advancement of the Science of Digital Games.

Hovy, E., and Lin, C.-Y. 1998. Automated text summarization and the summarist system. In *Proceedings of a workshop on held at Baltimore, Maryland: October 13-15, 1998*, 197–214. Association for Computational Linguistics.

Jones, S. 2014. Your favorite songs, abridged.

Khalifa, A. 2019. Mario AI framework 10th anniversary edition.

Lambe, I. 2016. Steam trailers in 6s @microtrailers.

Liapis, A.; Yannakakis, G. N.; and Togelius, J. 2014. Computational game creativity. In *Proceedings of the Fifth International Conference on Computational Creativity*.

Lienhart, R.; Pfeiffer, S.; and Effelsberg, W. 1997. Video abstracting. *Communications of the ACM* 40(12):54–62.

Liu, W.; Luo, X.; Zhang, J.; Xue, R.; and Xu, R. Y. D. 2017. Semantic summary automatic generation in news event. *Concurrency and Computation: Practice and Experience* 29(24):e4287.

Mawhorter, P., and Mateas, M. 2010. Procedural level generation using occupancy-regulated extension. In *Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games*, 351–358. IEEE.

Miaoqi Zhu, Fan Zhao, X. F., and Moser, C. 2017. Developing playability heuristics based on nouns and adjectives from online game reviews. *International Journal of Human–Computer Interaction* 33(3):241–253.

Mindek, P.; Čmolík, L.; Viola, I.; Gröller, E.; and Bruckner, S. 2015. Automatized summarization of multiplayer games. In *Proceedings of the 31st Spring Conference on Computer Graphics*, 73–80.

Nintendo. 2017. Super mario odyssey.

Panagiotopoulos, G.; Giannakopoulos, G.; and Liapis, A. 2019. A study on game review summarization. In *Proceedings of the Workshop MultiLing 2019: Summarization Across Languages, Genres and Sources*, 35–43.

Peeters, G.; La Burthe, A.; and Rodet, X. 2002. Toward automatic music audio summary generation from signal analysis. In *ISMIR*.

Shigeru, M.; Takashi, T.; Toshihiko, N.; and Kazuaki, M. 1985. Super mario bros.

Spiel, K.; Haimson, O. L.; and Lottridge, D. 2019. How to do better with gender on surveys: A guide for hci researchers. *Interactions* 26(4):62–65.

Spina, D.; Trippas, J. R.; Cavedon, L.; and Sanderson, M. 2017. Extracting audio summaries to support effective spoken document search. *Journal of the Association for Information Science and Technology* 68(9):2101–2115.

SteamLabs. 2019. Experiment 001 micro trailers.

Summerville, A. J.; Snodgrass, S.; Mateas, M.; and n'on Villar, S. O. 2016. The vglc: The video game level corpus. *Proceedings of the 7th Workshop on Procedural Content Generation*.

Thominet, L. 2016. 10/10 would review again: Variation in the player game review genre. *Technical Communication Quarterly* 25(3):167–177.

Wang, J., and Wang, H. 2012. *Cozy Classics: Pride & Prejudice*. Simply Read Books.

Wiki, S. M. 2020.