Discerning Human and Procedurally Crafted Content for Video Games

Tommy Thompson[1] and **Rob Watling**[2]

Abstract. We discuss the results of a preliminary study where participants discern between human and computationally crafted content for a video game. Participants were tasked with completing a portion of the game with the knowledge that segments were created either by a procedural generation algorithm or by a game designer. When asked to discern which segments were built by humans and vice versa, overall accuracy of participant guesses is relatively low. However, rationale reached by participants in making these conclusions leads to some interesting discussion about expectations of procedural generation systems and requirements for future studies.

1 Introduction

Procedural Content Generation (PCG) is a popular design paradigm found in video game development. While the origins of this method can be found in the likes of *Elite* [3] to overcome hardware limitations, the emphasis has shifted towards experimentation and challenge. This is typified by the *Borderlands* series [5]: where weapons and tools are presented for the player to discover, adopt or discard based upon personal preference. Meanwhile, *Diablo* [4] and *Spelunky* [21] adopt PCG for map generation in an effort to retain variety, novelty and challenge for even the most seasoned of players

If we consider this transition of the role of PCG systems, what is most interesting is that players perception of in-game content is becoming of greater focus. As problem scope increases, developers place a stronger emphasis on ensuring content is as interesting as it is varied. This has resulted in significant work in Artificial Intelligence (AI) to create intelligent PCG processes [19], with efforts to create 'custom' and more bespoke content [6, 20] and tools to aid the development process [8].

In this paper, we discuss preliminary work in generating content for an 'endless runner' game entitled *Sure Footing*[3]. The game tasks players with navigating a hazardous environment for as long as possible. Players are presented an early build of the game that carries content designed both by the developers and an early build of a PCG system. The task for participants was to identify the human-built and PCG samples and give a rationale for why they reached their conclusion. Our hypothesis was that if we were to base our PCG system on a meta-creative approach; adopting principles from a human designer, that players by-and-large would struggle to identify any key differences.

[1] University of Derby, UK, email: tommy@t2thompson.com
[2] University of Derby, UK, email: therobwatling@gmail.com
[3] A game being developed by Table Flip Games Ltd.: `http://www.tableflipgames.co.uk`

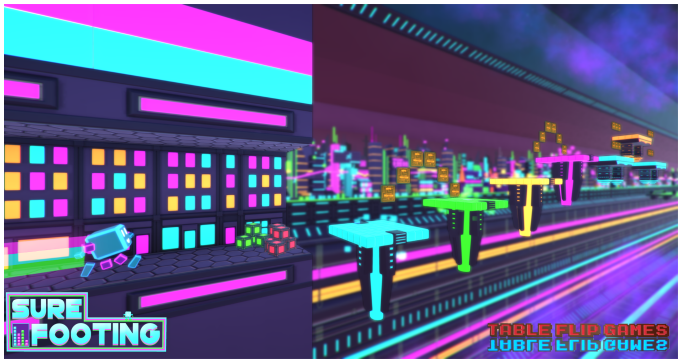

Figure 1. A screenshot of the *Sure Footing* video game, where the player, represented by a blue cube, must navigate a series of platforms and environmental hazards.

2 Sure Footing & Endless Runner Games

Sure Footing, shown in Figure 1 is an 'endless runner', where the player must navigate through a hazardous environment for as long as possible. Player's must traverse a collection of platforms and avoid obstacles placed upon them whilst evading an enemy that is following them throughout. Should the player fail a jump between platforms or be captured by their pursuer, the game will restart from the beginning of the current segment of play.

The endless runner genre is an effective platform for experimenting in PCG given that players are seldom aware of what is ahead of them. This allows for sudden change to the world that the player must adapt to. This is part of the novelty and charm that drove the popularity of seminal endless runner *Canabalt* [13] and subsequently titles such as *Flappy Bird* [10], and *Temple Run* [7].

Endless runners have a difficult balance to attain due to their unpredictable nature: should changes prove too sudden, players may subsequently lose interest. Ultimately, it is crucial that players feel the challenge of the game comes from their own ability to master game mechanics, rather than unfair design of the game. Equally players should be able to understand how to proceed through the game, irrespective of whether particular 'chunks' of level design have previously been seen in play. As discussed in Section 5, we place an emphasis on difficulty and progression in each participant's playthrough.

3 Related Work

Arguably the most established research in PCG for platforming games can be found in the *Mario AI Competition* which ran from 2009 to 2012 and has since been succeeded by the *Platformer AI Competition*[4]. The competition is dependent upon participants adopting a clone of the popular *Super Mario Bros.* [11] series. While originally intended to focus on gameplay, a level generation track was introduced in 2010 [18], with each entrant required to adopt player data from the an initial test level [14]. While the emphasis is to generate an intelligent and customised level generator, the focus of the competition is to find levels that judges deem 'interesting', rather than accurately reflect the designs of the *Super Mario Bros.* series. As such, the competition refrains from having judges compare PCG levels to original *Super Mario* levels built by human designers.

This work, among others in the AI field, focusses on search-based procedural generation. While this is an intelligent process that aims to create customised and unique content, there is seldom any emphasis on modelling the creative processes adopted by human designers in game development [2]. There have been notable exceptions to this, with one of the most prominent examples being the 'Sentient Sketchbook' project. As detailed in [8, 12, 9], this project carries a stronger emphasis on the use of PCG for human-designers as a tool; allowing for intelligent and useful content to be created in line with a designers expectations and habits.

The inspiration for this project is the *Tanagra* project detailed in [17]: a mixed-initiative design tool that aids in the creation of levels for 2D platformer games. The system allows for a designer to establish a timeline of 'beats': setting the pace of gameplay. The first phase of this work detailed in [15] is adopted in this project, where levels are built courtesy of rhythm groups which establish activities that take place.

4 System Design

In this section we give a brief overview of the PCG system adopted for this experiment. As we continue to discuss the design behind this system, we adhere to the taxonomy for PCG techniques defined in [19].

As noted in Section 3, our level generator adopts the rhythm approach discussed in [15]. The generator adopts a generate and test approach: creating and refining the rhythm of play followed by the geometry. The rhythm generator is comprised of a grammar representing player actions. This is encompassed by what is referred to as a *sprint*, a vector of game actions that lasts no longer than 60-90 seconds in-game. Actions are constrained to particular durations, denoted as *short* ($\leq$ 1 second), *normal* ($1-3$ seconds) or *long* ($3-5$ seconds). A full list of all available actions can be found in Table 1.

Once a full sprint vector is established, a critic will briefly evaluate to ensure a sense of flow is retained: the critic may swap pairs of activities, or add segments to give players a brief respite. This vector is passed into the geometry generator to create the level for play. This geometry generator is responsible not only for the selection of geometry but its subsequent placement within the game scene.

Each of the activities identified in Table 1 have one or more prefabricated pieces of geometry, hereby referred to as *prefabs*, that effectively represent the intended behaviour from the player. An example of this can be seen in Figure 2, which is one of the 'hopscotch' prefabs. The geometry generator places these items into the scene, aligning them such that a complete level is constructed. Once a sprint is completed, a 'rest' prefab is placed into the world. Typically this whole procedure is an online process and takes place during play. However, as discussed in Section 5, this process is made offline for the duration of this experiment.

[4] http://www.platformersai.com/

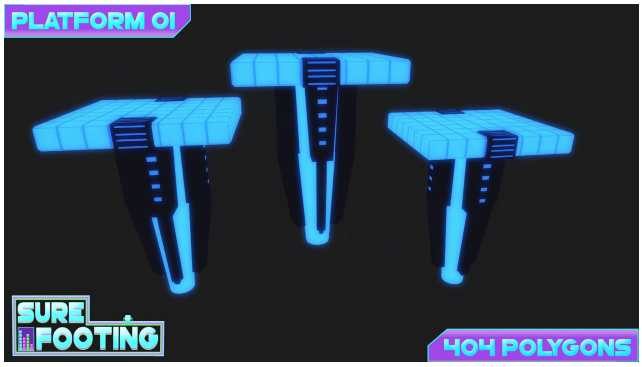

Figure 2. One of the prefab geometry pieces adopted by the geometry generator for the 'hopscotch' activity in Table 1.

5 Experiment Design

Our experiment was conducted during the *GameCity* festival in Nottingham, UK[5]. The focus of the experiment was to determine whether users could differentiate between levels crafted by a prototype PCG system, versus levels designed by one of the authors. In an effort to prepare for the festival, we exported six levels from the PCG system and stored them for later use. In addition to the PCG levels, six levels of equivalent length were crafted in the game engine by one of the authors.

While each level that was designed was unique, there are similarities that can be seen throughout. This is in part due to the prefabs discussed in Section 4 which were adopted in all level creation. In addition, given that the PCG system detailed in Section 4 was written by one author, with the other responsible for building the human levels, there is an argument to be made in that design habits of the authors have been injected, albeit rigidly, into the rhythm system. We return to these points in Section 6.1 and note the limitations they present as well as future steps for improvement.

Table 2. A breakdown of the percentage of participants who guessed either human or PCG-crafted level after each stage of completion. Followed by the success rates of those guesses at that particular stage.

Breakdown of Designer Guesses			
Level	Human Level	PCG-Level	Unsure
1	63.15%	23.7%	13.15%
2	50%	23.7%	26.3%
3	28.9%	42.1%	29%
Success Rates			
1	71.43%	25%	N/A
2	92.86%	12.5%	N/A
3	28.57%	37.5%	N/A

Each play-through of *Sure Footing* comprised of three 'levels'. With a minimum of one human and one PCG-crafted level per play-through. The third and final level was selected at random from the

[5] The festival took place during 25th October to 1st November 2014: http://www.gamecity.org

Action	Duration	Description
Run	Short, Normal, Long	A flat section of terrain which the player must run across.
Jump	Short	A gap between platforms which may carry a variation in height, such that can either jump or fall depending upon the context.
Incline	Normal	A series of short platforms closely placed to one another or a ramp that gradually increases in height.
Decline	Normal	A series of short platforms closely placed to one another or a ramp that gradually decrease in height.
Hopscotch	Normal	A series of short platforms with one in the middle that is higher than the others, forcing the player to hop atop or over it.
Fall	Normal, Long	Two platforms with separated by a significant vertical drop. Players are expected to fall or jump down to the lower platform.
Spring	Normal, Long	A long platform with a spring attached to the end that will launch the player to a much higher platform.

Table 1. The collection of actions that can take place in a given 'sprint' of play.

Table 3. A table showing the frequency of reasons left by participants. Including the percentage of responses that left a given reason, followed by a breakdown with respect to whether they guessed a level was human or PCG-crafted.

	Reasons For Decision						
	Difficulty	**Pace**	**Variety**	**Length**	**Item Placement**	**Don't Know**	**Other**
All Responses	35.09%	36.84%	29.82%	14.91%	29.82%	7.89%	8.77%
No Vote	0.88%	2.63%	0.88%	0.88%	1.75%	4.39%	3.51%
Decided Human-Crafted Level							
All Guessed Human	18.42%	23.68%	14.04%	10.53%	19.30%	0.88%	0.88%
Correctly Guessed Human	10.53%	7.89%	7.02%	5.26%	7.89%	0.88%	0%
Decided PCG-Crafted Level							
All Guessed PCG	15.79%	10.53%	14.91%	3.51%	8.77%	2.63%	4.39%
Correctly Guessed PCG	7.89%	4.39%	7.02%	2.63%	3.51%	0.88%	1.75%

PCG and human-designed sets, thus certain users would be exposed to each type of content, with one type more-so than the other.

At the beginning of the play-through, players were briefed that they would play at minimum one of each kind of level and that their task was to discern between the two types. Upon completion, the next level was immediately loaded into the game for the player to complete. In the event that players found these levels too challenging, the option was given to allow for a level to be skipped. Players were given as many tries as was necessary to complete the set of three levels. Upon completion, participants we asked if they could identify PCG and human samples; identifying whether level difficulty, pace, variety of rhythm, length and placement of items informed their decision. In addition, players were also given the option to express in detail additional elements that helped cement their opinion. Only after this questionnaire was completed and the game saved performance data was it revealed to users whether a given level was indeed crafted by a human or PCG system.

6 Results & Discussion

The results from 45 participants can be seen in Table 2, showing the breakdown of guesses at each stage of the process. In addition, we provide a breakdown of the frequency that particular reasons were given and their success in Table 3.

There are a number of interesting results, noting not only gradual trends in guessing patterns, but also the reasons given in certain circumstances. Firstly, we note that players were more likely to correctly denote a level as being crafted by a human than by the PCG system. This is perhaps not surprising, given that players would assume by default that content was man-made if they found it fun or engaging. Another interesting element is that not only is the success rate for voting PCG-levels less accurate, but players are more likely to be left unsure in their decision. Despite the level of accuracy behind human guesses, players became less confident over time in voting for a human-designed level, arguably due to not discovering a significant difference in the content that was being shown during gameplay. We believe this could be a limitation of the current generator, given PCG levels may appear remarkably similar to human-crafted content.

If we look further at the feedback from Table 3, it is interesting to note that that pace and difficulty followed by variety and item placement are deemed the biggest factors for making a given decision. Despite this, in certain circumstances this proved to be an incorrect assertion. For example, less than half of all participants who blamed pace for a human-designed level were proven correct. Overall, there does not appear to be a real consensus from this study for understanding whether a level was human or PCG-crafted.

In addition to the provided reasons, there was written feedback that was provided through the 'Other' column of the questionnaire. This yield some equally interesting yet contradictory reasons for participants decisions. Specific written feedback from participants noted that levels were "very good" or "intriguing", with several participants noting "flow" as one of the reasons for human-crafted samples, only to be proven wrong. One participant went so far as to criticise the design of one level, noting that "no human would place" a particular

segment of prefabs together and was correct in that assertion.

We note that the average success rate was 25%, with 29% of participants failing to recognise *any* level successfully. Meanwhile 13% were capable of scoring 100% accuracy, identifying all PCG and human-crafted levels. It is arguably their written feedback or experience that proved most valuable. One participant was an independent game developer who could 'see' the patterns at play. Meanwhile another noted that item placement in particular showed an emphasis on human design. Given blocks and power-ups would be dropped in what they deemed "easier" segments of play. One fact that is not made visible in Table 2 is that in two cases, participants completely ignored the briefing given to them and stated that all levels were man-made. We would argue that part of this challenge in the eyes of players originates in the problem domain. As discussed in Section 2, the endless runners constrain the amount of change available to the designer. In addition, there are still numerous limitations in our system which we will now discuss.

6.1 Study Limitations

While this study does yield some interesting results, there are some notable limitations both with the study as well as the current generation system that we aim to address in future studies.

Firstly, the *Sure Footing* generator is a weak computationally creative system [1]: given it is largely reliant upon the pre-conceived notions of the human authors. Art assets are stored in pre-built chunks the system is reliant upon and the generator is not overly flexible. As such, any level built will carry heavy influences from human designers. More importantly, this generator was not particularly expressive, with only differing configurations of one base level 'template' that could be achieved. While the range of expression permitted to the generator must be improved, relating back to our previous point, future studies must also focus on measuring the full expressivity of the system. This notion, as discussed in [16], can help us identify the range of content the generator can establish and subsequently what impact this has on player perceptions. In addition, this would allow for assessment of whether current generators can build the same range of content as a human designer.

Furthermore, future studies would benefit from multiple generators for players to consider: ranging from humans, to intelligent procedural generations systems, with a variety of purely random generators in between. Lastly, future studies would benefit from testers being able to identify particular areas of gameplay where their suspicions of PCG or human-driven design are raised.

7 Conclusion

In this paper we highlighted a short study assessing players perceptions of procedurally generated versus human-crafted content for an endless-runner game. Players proved more successful in identifying human-crafted content than one by a PCG system, which in some respects is a positive step for the level generator; given that the majority of players could not find any patterns or trends that identified a given sample as procedurally generated. Given that this generator is influenced by a human creative process, it is perhaps to be expected that players find it harder to identify PCG-crafted levels. However, when we consider that the PCG system is rather rigid in this current version, it is surprising that the majority of users do not identify any real differences.

The feedback from this process has been adopted by the *Sure Footing* team who aim to build an improved level generator. Future work is focussed on building a more intelligent solution, in addition to addressing the issues raised in Section 6.1, such that a second study may be conducted over a longer period. This would allow for richer discussion of players perceptions of procedurally generated content as the generator becomes more expressive and their restrictions lifted.

ACKNOWLEDGEMENTS

The authors would like to thank the *Sure Footing* development team: Jonathan Boorman, Neall Dewsbury, Charlotte Sutherland, Matthew Syrett and James Tatum, for their assistance with this study.

REFERENCES

[1] Mohammad Majid al Rifaie and Mark Bishop, 'Weak and strong computational creativity', in *Computational Creativity Research: Towards Creative Machines*, 37–49, Springer, (2015).
[2] Nuno Barreto, Amílcar Cardoso, and Licínio Roque, 'Computational creativity in procedural content generation: A state of the art survey', in *Proceedings of the 2014 Conference of Science and Art of Video Games)*, (2014).
[3] Bell, I. and Braben, D. Elite. Acornsoft, 1984.
[4] Blizzard North. Diablo. Blizzard Entertainment, 2009.
[5] Gearbox Software. Borderlands. 2K Games, 2009.
[6] Erin J Hastings, Ratan K Guha, and Kenneth O Stanley, 'Evolving content in the galactic arms race video game', in *Computational Intelligence and Games, 2009. CIG 2009. IEEE Symposium on*, pp. 241–248. IEEE, (2009).
[7] Imangi Studios. Temple Run, 2011.
[8] Antonios Liapis, Georgios N Yannakakis, and Julian Togelius, 'Sentient sketchbook: Computer-aided game level authoring.', in *Proceedings of the 8th International Conference on the Foundations of Digital Games*, pp. 213–220, (2013).
[9] Antonios Liapis, Georgios N Yannakakis, and Julian Togelius, 'Designer modeling for sentient sketchbook', in *Computational Intelligence and Games (CIG), 2014 IEEE Conference on*, pp. 1–8. IEEE, (2014).
[10] Nguyen, D. Flappy Bird. .GEARS Studios, 2013.
[11] Nintendo EAD. Super Mario Bros. Nintendo, 1985.
[12] Mike Preuss, Antonios Liapis, and Julian Togelius, 'Searching for good and diverse game levels', in *Computational Intelligence and Games (CIG), 2014 IEEE Conference on*, pp. 1–8. IEEE, (2014).
[13] Saltsman, A. Canabalt. Semi-Secret Software, 2009.
[14] Noor Shaker, Julian Togelius, Georgios N Yannakakis, Ben Weber, Tomoyuki Shimizu, Tomonori Hashiyama, Nathan Sorenson, Philippe Pasquier, Peter Mawhorter, Glen Takahashi, et al., 'The 2010 mario ai championship: Level generation track', *Computational Intelligence and AI in Games, IEEE Transactions on*, **3**(4), 332–347, (2011).
[15] Gillian Smith, Mike Treanor, Jim Whitehead, and Michael Mateas, 'Rhythm-based level generation for 2d platformers', in *Proceedings of the 4th International Conference on Foundations of Digital Games*, pp. 175–182. ACM, (2009).
[16] Gillian Smith and Jim Whitehead, 'Analyzing the expressive range of a level generator', in *Proceedings of the 2010 Workshop on Procedural Content Generation in Games*, p. 4. ACM, (2010).
[17] Gillian Smith, Jim Whitehead, and Michael Mateas, 'Tanagra: A mixed-initiative level design tool', in *Proceedings of the Fifth International Conference on the Foundations of Digital Games*, pp. 209–216. ACM, (2010).
[18] Julian Togelius, Noor Shaker, Sergey Karakovskiy, and Georgios N Yannakakis, 'The mario ai championship 2009-2012.', *AI Magazine*, **34**(3), 89–92, (2013).
[19] Julian Togelius, Georgios N Yannakakis, Kenneth O Stanley, and Cameron Browne, 'Search-based procedural content generation: A taxonomy and survey', *Computational Intelligence and AI in Games, IEEE Transactions on*, **3**(3), 172–186, (2011).
[20] Geogios N Yannakakis, 'Game ai revisited', in *Proceedings of the 9th conference on Computing Frontiers*, pp. 285–292. ACM, (2012).
[21] Yu, Derek. Spelunky. Mossmouth, 2009.