Procedural Generation of Game Maps with Human-in-the-Loop Algorithms

Levi H. S. Lelis, Willian M. P. Reis, and Ya'akov (Kobi) Gal

***Abstract*—A key challenge in procedural content generation is to automatically evaluate whether the generated content has good quality. In this paper we describe an approach that uses non-expert workers to evaluate small portions of levels generated by an off-the-shelf generation system for the game of Infinite Mario Bros. Several such evaluated portions are then combined to form full levels of the game using a mathematical progression arc model. The composition of the small portions into full levels is done by accounting for the human-annotated information. We evaluated the approach using computational metrics as well as surveying human subjects playing the levels. The results show that the human computation approach is able to generate levels that are perceived by people to have better visual aesthetics and to be more enjoyable to play than existing approaches. Another contribution of our work is a dataset of the small annotated levels that can be used in future research for learning models for evaluating machine generated content.**

***Index Terms*—Procedural Content Generation, Platform Games, Human Computation**

I. Introduction

PROCEDURAL Content Generation (PCG) is an approach for automatically generating content for specific problem domains. For example, when applied to computer games, PCG systems automatically produce levels, rules, textures, and other contents that are traditionally created by human professional designers [1]. The PCG approach has great potential as it can be used to generate content that is tailored to the styles of individual people [2], [3]; it may increase engagement by providing a new experience for players every time they play a game [4]. On the other hand, it is difficult to evaluate the quality of the generated content. Specifically, PCG can generate a large number of candidate game components such as levels and textures, and using game designers to evaluate such components can be time consuming and expensive.

The contribution of this work is a novel PCG approach that addresses the challenge above by combining a human-in-the-loop approach with a mathematical model of difficulty progression. In our human-in-the-loop approach we rely on non-expert human workers to measure whether particular content is of good or bad quality. Human evaluations can be quickly obtained from crowdsourcing environments such as Amazon Mechanical Turk [5] (AMT) and from students or volunteers. Our approach uses an existing PCG system to generate thousands of small units of play called "segments". We use human computation [6] to evaluate these segments for their enjoyment, visual aesthetics, and difficulty. We combine the individual segments to a complete level by accounting for the human-annotated information and a simple mathematical model of difficulty progression [7].

We applied this approach to *Infinite Mario Bros* (IMB) [8], a clone of *Super Mario Bros*. Non-expert human workers annotated over one thousand segments of the IMB game with respect to the workers' perceived enjoyment, visual aesthetics, and difficulty. These annotated segments were subsequently combined to form complete levels of the game. We conducted two user studies to evaluate our approach. In the first study, we established that people preferred playing a level composed of segments of increasing difficulty to a level composed of segments of random difficulty. In the second study, users reported that the levels generated by the human computation approach were perceived to be more enjoyable to play and to exhibit better visual aesthetics than the levels generated by the other approaches. We have made the database of annotated segments publicly available.[1] Our results demonstrate that intelligently combining opinions from non-experts can provide a novel PCG approach that is able to generate visually pleasing and enjoyable levels of a platform game.

This work extends a preliminary publication [9]. First, we analyze how the number of segments each worker evaluates in a single session of play influences the rating values (see Section V-D). Second, we analyze how our approach changes the expressive range [10] of the evaluated segments after combining them with a mathematical model of difficulty progression (see Section VIII). Third, we increase from 37 to 53 the number of subjects who participated in the user study that compares our PCG approach with previous methods.

The paper is organized as follows. In Section II we review the relevant literature. In Section III we present relevant background material. In Section IV through VI we explain our human-computation approach. In Section VII we present the results of a detailed user study we conducted to evaluate our system. In Section VIII we present the expressive range study of our approach.

II. Related Work

Our work relates to the research encountered in two separate areas: procedural content generation and human computation. In this section we review relevant works in each of these areas.

A. Procedural Content Generation

PCG algorithms have been applied to different game genres, such as dungeon crawlers [11], real-time strategy games [12],

L. H. S. Lelis and W. M. P. Reis are with the Departamento de Informática at Universidade Federal de Viçosa, Viçosa, Brazil.

Y. Gal is with the Department of Software and Information Systems Engineering at Ben-Gurion University of the Negev, Beer-Sheva, Israel.

[1] http://www.dpi.ufv.br/~lelis/downloads/Mario-Dataset.zip

physics-based puzzles [13], racing [14] and arcade games [15], platform games [16], and other genres.

Smith et al. [17] presented Tanagra, a system for developing levels for 2D platform games such as IMB. Tanagra allows the game designer to specify parts of the level and the system completes the level while respecting the designer's decisions. Sorenson et al. [18] presented a system which uses the idea of rhythm groups introduced by Smith et al. [10] to define a computational model of player enjoyment to evolve levels of IMB. Sorenson et al.'s system is similar to our approach in that it uses a mathematical model to generate content, but it does not incorporate human input in the process.

Shaker et al. [2] describe a system for generating player-specific content for IMB which directly asks each player to evaluate the generated content during his or her interaction. This approach disrupts the user's play, which is time consuming and may deter the player's enjoyment. By contrast, our system uses human workers as a pre-processing step without disrupting individual players during their play.

Many have used computational models to procedurally generate content for games, but without involving non-expert workers in the loop (e.g., [19], [20]). Two formidable examples include Shaker et al. [21], who showed how to extract features to learn predictive models of the player's experience in IMB; Snodgrass and Ontañón [22] introduced a PCG system that learns Markov chains from SMB levels created by professional designers, which are then used to generate novel game levels.

We describe work on automatically evaluating the generated content in computer games for players' enjoyment. For example, Togelius et al. [23] use a player's behavior to generate race tracks which are more fun to the player; Liapis et al., [24] introduce general evaluation functions which are applicable to different games; Sorenson et al. [18] learn a model of enjoyment based on levels generated by professional designers. Such works are usually motivated by the fact that it is not possible to have humans evaluating content produced by machines. For example, Shaker et al. [25] stated that "because of the large amount of content that can be generated, it is not feasible to humanly judge the results, and automatic evaluation becomes a necessity". Our work shows that by using people to evaluate smaller segments and combining them intelligently, we are able to construct full-sized levels that are enjoyable to play without having to manually annotate the entire game.

Another line of research within PCG is concerned with the development of mixed-initiative systems. In such systems the game designer "takes turns" with the system in the creation process. For example, Sketchbook Sentient is a computer-aided design tool for creating game maps that operates on high-level sketches instead of detailed maps. A genetic algorithm based tool presents suggestions in real time, allowing the user to replace parts of the map being created [26]. Ropossum is also a mixed-initiative system that creates complete and playable levels of Cut the Rope and allows the game designer to fine adjust the content created [27]. Butler et al. also introduced a mixed-initiative system to generate levels for the educational game of Refraction [28], which allows the designer to define a progression plan which is subsequently satisfied by the computer. Campos et al. [29] presented a mixed-initiative system that allow the designer to sketch the level of a physics-based game with a drawing tool and the system creates the level's structure from the sketch. These systems differ from our approach because they were built to assist game designers. By contrast, our system aggregates the feedback of non-professional workers to build game levels.

B. Human Computation

Research on human computation investigates methods for using human intelligence to solve problems that computers cannot solve alone [30]. Examples abound, from labeling pictures on the internet [31] and classifying images from the Sloan Digital Sky Survey [32] to human-guided genetic algorithms [33]. Obtaining services and information by soliciting contributions from individual people is not a new idea, but the growth of online communities has made crowdsourcing platforms an easy way to obtain access to human intelligence on demand in a scalable way. These platforms include work-for-pay services such as Amazon Mechanical Turk, as well as volunteer-based and citizen science services [32].

More recently, there has been growing work in artificial intelligence that focuses on devising innovative ways to harness human intelligence efficiently. For example, the CrowdSynth effort [34] combines machine learning and decision-theoretic optimization techniques to manage hiring decisions and task allocation in crowdsourcing. They show how learned probabilistic models can be used to fuse human and machine contributions to predict the behaviors of workers. Other works have used AI methods to increase engagement in citizen science [35] or used decision theoretic planning and execution for managing workflow as an optimziation problem [36].

There are few works using the crowd to design and evaluate textual content that was automatically generated. Sina et al. [37] used planning algorithms to generate stories about daily activities from a pre-defined database and used the crowd to rate the resulting content for how believable and consistent they were. The work of Li et al. [38] generates a corpus of narrative examples using the crowd. They infer a graphical representation over story events that is subsequently sampled to create narratives using a fitness function. We are the first to use the crowd to generate game levels.

III. Background

A. The Infinite Mario Bros Domain

The game of IMB is a popular level-based game which also serves as a testbed for evaluating PCG systems in the literature. The advantage of using IMB in our study is that we are able to compare the quality of the content generated by our system with that of the state-of-the-art.

Three screenshots of IMB are shown in Figure 2. The player controls Mario (on the center of screen). Mario's goal is to reach the rightmost spot of the level. In order to succeed, Mario has to avoid enemies and other challenges. The IMB levels are grid spaces containing a set of objects such as mountains (the grass-topped boxes in the figure) and enemies taking the form of shooting cannons, turtles, and others. The number of these objects and their distribution on the grid

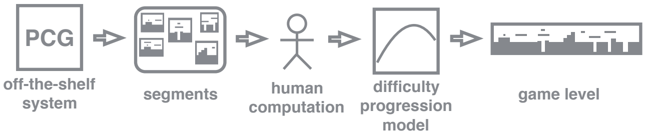


Fig. 1: Overview of our human-in-the-loop approach to PCG.

determines the difficulty of the level. Every object is associated with an (x, y) location on the grid and some of the objects such as mountains and pits can have different heights and widths—boxes, a few enemies, and the small version of Mario himself occupy a single cell on the grid. In this paper all full levels are represented as a grid of size 160×15.

IV. The Human Computation Approach to PCG

In this section we describe our Human-Computation PGC System, which we refer as HCS. A high-level description of this approach, shown in Figure 1, is as follows.

1) A PCG system generates a large library of small levels of IMB called "segments", that people can play and evaluate quickly.
2) Workers annotate each of the segments in the library with respect to three subjective measures: enjoyment, visual aesthetics, and difficulty.
3) The segments are combined using a mathematical difficulty progression model that accounts for the human-annotated values of the subjective metrics.

There are two main advantages to using the HCS approach: First, human workers employed by our system do not have to be knowledgeable in the craft of game design—workers evaluate the segments according to their perception as game players. A large number of annotated segments can be quickly obtained using volunteers or workers on crowdsourcing platforms. Second, we do not disrupt the player's gameplay as our approach only asks questions to human workers solving small portions of the game as a preprocessing step.

V. Segment Generation and Human Annotation

The first question to consider is the size of the segments workers will evaluate. Small segments allow for a quicker evaluation. However, the segments may be too short for people to reach definite conclusions about our subjective criteria. On the other hand, large segments may take too long to play and require considerable effort to generate a repository that is large and diverse enough for our approach. We decided on a grid size of 20×15, which, as we established in a pilot study, takes an average of 20 seconds for people to play. This allowed us to generate a library of approximately 2,000 segments without compromising the quality of evaluations, as we show in the empirical analysis.

The system we use for generating the library of segments, which we denote as Γ, was created by the game designer Markus "Notch" Persson [39] and is publicly available—Notch's system is often called Notch level generator (NLG). This system is a black box that follows a set of hardcoded rules to stochastically create segments of IMB. NLG receives as input a difficulty value d for stochastically determining the number of enemies in the segment. Higher values of d correspond to more challenging segments. NLG starts with an empty segment—an empty grid of size 20×15—and iteratively adds objects to the grid according to the value of d and to a set of design heuristics.

We used NLG to generate more than 2,000 segments of size 20×15 with values of d selected uniformly at random to ensure a collection of segments Γ with varying difficulty. The disadvantage to NLG is that due to the stochasticity of the generator, it may also produce segments which are not visually appealing and are not necessarily enjoyable to play. For example, the objects in the segment shown in Figure 2a are oddly placed on the screen: the block and pipe in the upper part of the screen has no purpose since they are not reachable by Mario. In addition, sometimes a segment l has a large number of enemies and challenges (determined by a large value of d) but l is not a difficult segment to play due to an alternative path that Mario can safely take in the game.

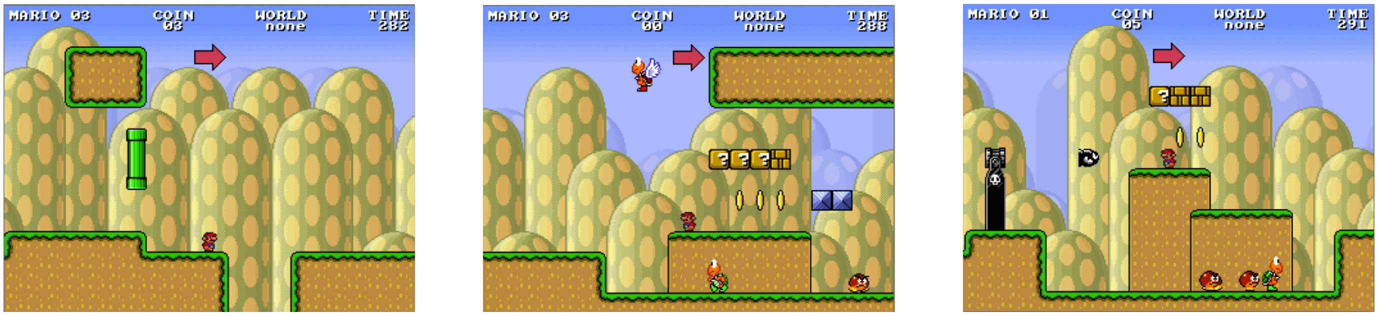
To address these challenges we enlisted human workers to play the segments in Γ and annotate each of the segments with respect to three measures: enjoyment, visual aesthetics, and difficulty. We made our system available for download and invited undergraduate and graduate students in the Departamento de Informática, at Universidade Federal de Viçosa, in Brazil, to play the segments. The students voluntarily and anonymously played the segments. Before playing the segments the volunteers were provided with a simple tutorial. The tutorial consisted of a set of instructions of the game control, and after reading the instructions, the volunteers played a sample segment. The volunteers were informed that the segments played would be much smaller and shorter than regular levels of the game of IMB. If some of the volunteers had played the game of Mario before, we did not want them to be disappointed by the reduced size of the segments.

Prior to evaluation, we preambled to the beginning of each segment a small grid of size 5×15 containing no objects but the ground level. Mario starts in this preamble, which allows the player to integrate naturally into the segment.

After playing each segment the volunteers were asked how much they agreed with the following statement: "I enjoyed playing the game", "I found the game to be visually pleasing", and "I found the game to be difficult". For each of these statements, the volunteers provided a score representing how much they agreed with each statement. The scores were provided in a Likert scale, i.e., they ranged from 1 (strongly agree) through 9 (strongly disagree). For example, a score of 1 for enjoyment, visual aesthetics, and difficulty means that the segment was perceived to be very enjoyable, very visually appealing, and very difficult.

The volunteers were presented segments in Γ in random order. This approach ensured that volunteers annotated most of the segments in Γ, and some of the segments were annotated more than once by different volunteers, in which case we consider the median scores. We removed from Γ the segments that were not annotated by any of the volunteers.

Since a segment is small enough to be played in a few seconds, a single volunteer could produce several annotated



(a) enjoyment:8, aesthetics:9, difficulty:7

(b) enjoyment:4, aesthetics:3, difficulty:5

(c) enjoyment:1, aesthetics:1, difficulty:4

Fig. 2: Three representative human evaluations for segments in Γ. These evaluations do not represent the authors' opinions, but show representative segments labelled by the volunteers who participated in our study.

segments in just a few minutes. In total the volunteers provided 2,715 evaluations of 1,437 distinct segments. These evaluations were obtained in 125 different sessions of play. A session of play is defined by a volunteer entering the system, annotating a collection of segments, and exiting the system. Note that the same volunteer could have multiple sessions of play. Since we wanted to maximize the number of annotated segments, in order to simplify the annotation process, we did not ask the volunteers for their identity and did not require the creation of a user account in our system. As a result, we do not know exactly how many volunteers participated in the experiment. However, we believe that the number of sessions of play offers a good upper bound on the number of distinct volunteers. We provide more details about the number of segments annotated in each session of play in Section V-D.

A. *Representative Annotated Segments*

Figure 2 shows three representative human annotated segments. The numbers in parenthesis show the value in a Likert scale, ranging from 1 to 9, for enjoyment, visual aesthetics, and difficulty, respectively. The segment shown in Figure 2a offers no challenge to the player and some of the objects are not even reachable, which might explain the segments' poor scores for enjoyment (score of 8) and visual aesthetics (score of 9); the segment was considered an easy one (score of 7).

The segment shown in Figure 2b was deemed as easy by a human worker (difficult of 5), despite the segment containing challenging enemies such as a flying enemy coming from above. The worker found the segment easy probably because they had enough time to reach the center part of the segment, when the flying enemy would cause no harm to the player. This example demonstrates that counting the number of enemies (the approach used by NLG to determine difficulty) may not always agree with the player's perceived difficulty.

The segment shown in Figure 2c poses an interesting and possibly enjoyable dilemma the player will face. In order to collect the power-up item the player becomes vulnerable to the bullets being shot by the cannon on the left-hand side of the segment. Alternatively to collecting the power-up item, the player could choose to quickly jump on top of the yellow blocks and safely advance to the next segment in the level.

B. *Correlation Results of the Annotated Segments*

We found a high positive correlation (coefficient 0.72) between enjoyment and visual aesthetics in the annotated segments. Similar results have been documented in other works. For example, in a study of web site aesthetics, Lavie and Tractinsky [40] reported high positive correlation (coefficient 0.68) between user reported aesthetics and pleasure. The strong correlation between visual aesthetics and enjoyment we observed in our experiment suggests that, in future works, one might wish to annotate the segments only with respect to the workers' enjoyment, as much of their perceived visual aesthetics might already be encoded in their reported enjoyment.

We also found a highly positive correlation (coefficient of 0.67) between difficulty and enjoyment. The Yerkes-Dodson law [41] states that enjoyment increases with difficulty up to some point, where the level gets too difficult to be enjoyable. The strong correlation between enjoyment and difficulty indicates that our volunteers tended to be skilled players who enjoyed playing the difficult levels of our library.

The correlation coefficient between visual aesthetics and difficulty was of 0.47, which is much smaller than the coefficient between enjoyment and visual aesthetics (0.72), and the coefficient between difficulty and enjoyment (0.67). This result makes sense, since one can easily imagine segments which are too difficult to play due to a large number of enemies and are visually unpleasing due to the poor placement of objects.

C. *Inter-Annotator Agreement*

Two independent volunteers agreed to contribute non-anonymously to our data collection. The 453 level segments these two volunteers evaluated in common allow us to perform an inter-annotator study by measuring the correlation of their annotations. The Spearman correlation values of these two annotators for difficulty, enjoyment, and visual aesthetics were 0.80, 0.45, and 0.38, respectively. Although with the caveat of only accounting for the annotations of two volunteers, these correlation values suggest that the difficulty annotations of a worker might be used to reliably predict the perceived difficulty of other people. The lower correlation values for enjoyment and visual aesthetics confirm the more subjective nature of these metrics. Yet, correlation values of 0.45 and 0.38 demonstrate a good level of agreement between the annotators,

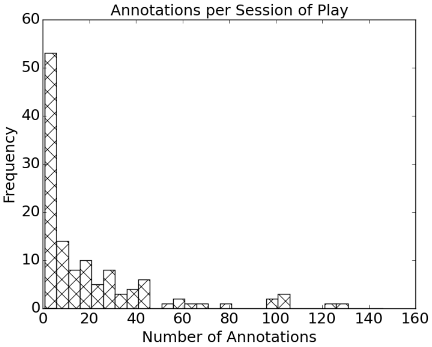


Fig. 3: Histogram of annotations per session of play.

suggesting that the enjoyment and visual aesthetics annotations of a volunteer might be used to approximate the enjoyment and perceived visual aesthetics of other players.

D. Analysis of Number of Evaluations Per Session of Play

In this section we analyze the volunteer segment ratings based on the number of segments that were evaluated. Figure 3 shows the histogram of the number of segments that were evaluated in a given session of play. As shown by the figure, volunteer contribution follows a long-tail distribution that has also been documented in other self-motivated domains such as citizen science [35]. Most volunteers evaluated fewer than 10 segments. The volunteer who evaluated the largest number of segments in a single session of play annotated 201 segments (not shown in the histogram, which capped the number of segments at 160 for readability purposes).

To study the differences between the ratings of "heavy" volunteers who evaluated many segments and "light" volunteers who evaluated few segments, we divided the evaluations into two groups, L (for light) and H (for heavy). Group L contains annotation sessions with at most 45 segments, and H contains annotations with more than 45 segments. The threshold of 45 segments was chosen such that L and H have nearly the same number of annotations: L has 1,377 while H 1,338.

Figure 4 shows the frequency of evaluations for each Likert value for enjoyment, visual aesthetics, and difficulty, respectively. We now discuss each figure in turn.

The rating distributions of L and H for enjoyment and visual aesthetics are similar in that they both express a unimodal distribution. In particular, volunteers in group L were more likely to rate segments as highly enjoyable and visually pleasing (values 1-3 on the Likert scale) than volunteers in group H. There are two possible reasons that can explain this. First, volunteers in H evaluated more segments than volunteers in L, making them more selective about which segments to attribute high scores. By contrast, volunteers in L tended to attribute the scores 1 and 2 to a larger number of segments. Another possible explanation is that volunteers in H may have reached a state of fatigue and boredom, thus reducing the number of segments they classified as enjoyable and visually pleasing. It may be that both factors contributed to volunteers in L assigning higher scores to a larger number of segments.

Algorithm 1 Progression-Arc Concatenation

```
Require: Set of segments $\Gamma = \{\Gamma_1, \Gamma_2, \cdots, \Gamma_9\}$, progression
    arc $T = \{d_1, d_2, \cdots, d_M\}$, integer $k$.
Ensure: Totally ordered set $\nabla = \{l_1, l_2, \cdots, l_M\}$
$\nabla \leftarrow \{\}$
for $i = 1$ to $M$ do
    $E_{k,d_i} \leftarrow k$ most enjoyable segments from $\Gamma_{d_i}$
    $V_{k,d_i} \leftarrow k$ most visually pleasing segments from $\Gamma_{d_i}$
    $l \leftarrow$ a random segment from $E_{k,d_i} \cap V_{k,d_i}$
    $\nabla \leftarrow \nabla \cup l$
```

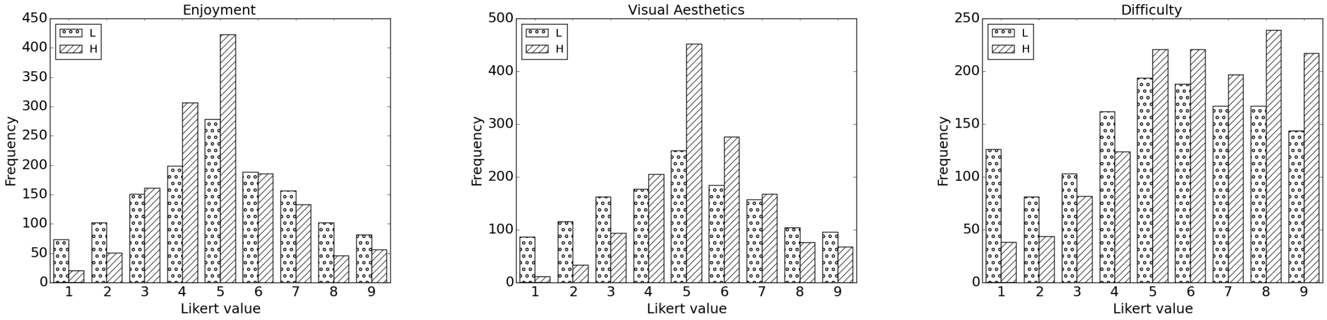
As for the rating distributions of difficulty, we observe in Figure 4c that volunteers in H classify the segments as easy (scores of 8 and 9) much more often than they classify the segments as difficult (scores of 1 and 2). We conjecture that volunteers increase their skills as they annotate the segments. Since the volunteers in H played more segments, they likely achieved a level of skill not achieved by volunteers in L. Thus, a level that is easy for a volunteer in H might be considered more difficult for a volunteer in L.

VI. Combining Segments with a Model of Difficulty Progression

The human computation procedure described above outputs a collection of annotated segments. In this section we describe how HCS combines these segments into a full IMB level with grid size 160×15. A common approach in level design is to place the hardest challenges of a level toward its end [7]. In HCS we use a similar approach in which we create IMB levels by concatenating segments while following a progression arc that places the hardest segments toward the end of the level.

Let a $\nabla = \{l_1, l_2, \cdots, l_M\}$ be a sequence of segments with $\nabla \subseteq \Gamma$ and each segment $l \in \nabla$ with size $x \times y$. ∇ represents a complete level of size $(x \cdot M) \times y$ formed by the concatenation of the segments $l \in \nabla$ according to some ordering $\mathbf{O}$. The final output of HCS process is ∇. In HCS a *difficulty progression arc* T defines the ordering $\mathbf{O}$ of segments. T is a sequence of difficulty values $\{d_1, d_2, \cdots, d_M\}$ where d_1 is the difficulty of the first segment composing ∇, d_2 is the difficulty of the second segment, and so on. Figure 5 shows one of the progression arcs used in our study, which we call a parabolic progression arc. The x axis represents the segment index in ∇, as defined by the ordering $\mathbf{O}$. The y axis represents the human-annotated difficulty value. This progression arc follows a common design choice of having the difficulty of the level increase as the player goes through the level (see, for example, [7]). However, as explained before, aiming at preventing the creation of very challenging levels, after reaching the largest difficulty value (segments 5 and 6 in ∇), HCS concatenates segments of smaller difficulty (segments 7 and 8 in ∇).

HCS, shown in Algorithm 1, receives as input a collection of annotated level segments Γ, a progression arc T, and a threshold k. We divide Γ into disjoint subsets $\{\Gamma_1, \Gamma_2, \cdots, \Gamma_9\}$, where Γ_j contains all segments with difficulty value of j. Let $E_{k,j} \subseteq \Gamma_j$ be the set of k-best segments in Γ_i with respect to the workers' reported enjoyment. Similarly, let



(a) Histogram of User Enjoyment.

(b) Histogram of Visual Aesthetics.

(c) Histogram of Difficulty.

Fig. 4: Frequency of evaluations for each Likert value for enjoyment, visual aesthetics, and difficulty.

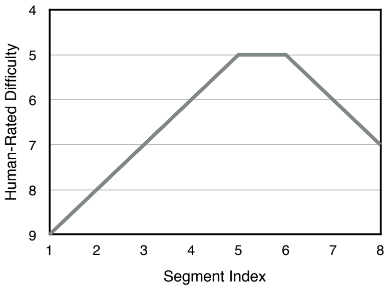


Fig. 5: Parabolic progression arc used in our experiments. The difficulty values are obtained with human computation, and the values on the x-axis denote the arc's ordering.

$V_{k,j} \subseteq \Gamma_j$ be the set of k-best segments with respect to the workers' perceived visual aesthetics. The level ∇ is composed by segments in $E_{k,j} \cap V_{k,j}$ of subsets Γ_j (see lines 3–5 of Algorithm 1). We constrain the composition process to segments that are rated highly with respect to both enjoyment and visual aesthetics by workers.

The value of k might have a large effect on the way the segments combine to form a full-size level. Larger values of k allow for a larger set of segments to choose from, but can result in choosing individual segments of low quality which may also affect the quality of the full level. Small values of k tend to provide higher quality segments, but there may be few segments to choose from, which can lead to repetition of segments in the full-size level and harm the player's enjoyment. In our experiments we use $k = 50$ which was tuned according to a pilot study.

After creating a full level we perform a post-processing step to deal with possible adjacent segments whose ground heights do not match. This is done by applying the "green topping" where height mismatches occur. Figure 6 shows a level generated by HCS using the parabolic progression arc shown in Figure 5. The challenges the player faces increase toward the end of the level, as described by the arc.

VII. Empirical Evaluation

In this section we describe the user study we conducted to evaluate the HCS system. We evaluate four systems: HCS with the progression arc shown in Figure 5 (denoted HCS-P, where the P stands for "parabolic", the shape of the arc), HCS with a random arc (denoted HCS-R and explained below), the Occupancy-Regulated Extension generator (ORE) [42] which was the winner of the 2011 Mario AI Competition, and NLG.

The progression arc implemented for the HCS-R approach returns a random integer in the interval $[5, 9]$, which is the same difficulty range used by HCS-P (cf. Figure 5), for each segment composing the full-sized level. We use HCS-R as a baseline method for testing if the commonly used level design approach of increasing the segments' difficulty indeed results in more enjoyable levels than a simple random baseline.

We carried out two user studies. In the first we compare HCS-P with HCS-R (Experiment 1), in the second we compare HCS-P with NLG, and ORE (Experiment 2); as we show below, HCS-P performed better than HCS-R in terms of enjoyment, and that is why we use it in Experiment 2.

A. Hypotheses

We are interested in evaluating if HCS is able to generate levels that are perceived to be visually pleasing and enjoyable to play. Specifically, we test the following hypotheses:

- **H1** HCS-P generates levels that are perceived to be more enjoyable to play than the levels generated by HCS-R.
- **H2** HCS-P generates levels that are perceived to be equally visually pleasing as those generated by HCS-R.
- **H3** HCS-P generates levels that are perceived to be more enjoyable to play than those generated by NLG and ORE.
- **H4** HCS-P generates levels that are perceived to be more visually pleasing than those generated by NLG and ORE.

B. Methodology

1) Evaluated Metrics: The systems were evaluated according to the following criteria: enjoyment, visual aesthetics, and difficulty. Each participant was asked to answer how much they agreed or disagreed, in a 7-Likert scale,[2] with the following affirmatives: (1) this level is enjoyable to play; (2) this level has good visual aesthetics; (3) this level is difficult.

[2] A few workers were confused by the 9-Likert scale used in the data collection experiment and attributed for difficulty a score of 1 for a segment that was clearly a 9. We decided to use a 7-Likert scale in the evaluation of the systems hoping that with fewer options there would be less confusion.

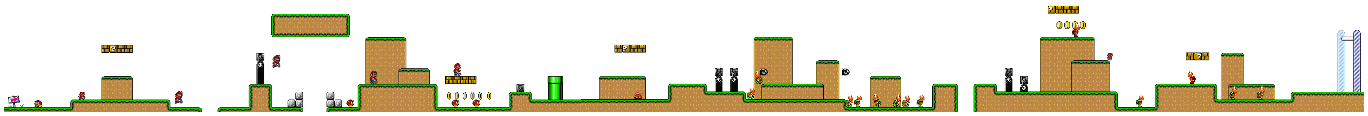


Fig. 6: A representative full-sized level generated by HCS using the parabolic progression arc.

A score of 1 for enjoyment and visual aesthetics mean that the participant strongly agrees that a level l is enjoyable and has good visual aesthetics; a score of 1 for difficulty means that the participant strongly agrees that l is difficult.

An alternative design for eliciting users' satisfaction would be to ask to play the level in all systems and then to rank them (e.g., l_1 is more enjoyable than l_2). We preferred a rating approach to a ranking approach for the following reasons. First, because each full-sized level took several minutes to play, we wanted participants to rate each level immediately after playing them. Waiting to play all systems before rating them requires considerable cognitive effort and may result in memory bias. Second, it is easier to train machine learning models to predict the satisfaction of users when they are measured as values, rather than a complete ranking. As an example, Guzdial et al. [43] have used part of our data to train a convolutional neural network. Lastly, we note that it is possible to convert our ratings data into ranking data as suggested by Yannakakis and Hallam [44].

One disadvantage of using rating, as demonstrated by the experiments of Yannakakis and Hallam [44], is that the results might suffer from ordering effects. To counteract such effects we employ a balanced Latin square design to try to ensure that the systems are tested the same number of times for each possible "playing position". For example, if we test 3 systems with n participants, and every participant sequentially plays one level of each system, a balanced Latin square ensures that every system is tested $n/3$ times as the first system played by the participant, $n/3$ as the second, and $n/3$ as the third. Naturally, depending on the number of participants the division of tests might not be perfectly balanced.

2) Participants: We used a within-subject study design for both experiments. Experiment 1 had 37 participants: 32 males and 5 females with an average age of 23.95 and standard deviation of 4.48. Each participant played one level generated by each system (HCS-P and HCS-R), resulting in the evaluation of 37 levels of each PCG system. Experiment 2 had 53 participants: 43 males and 10 females with an average age of 25.18 and standard deviation of 5.51. Each participant played one level generated by each system (HCS-P, NLG, and ORE), resulting in the evaluation of 53 levels of each PCG system. The number of participants reported in both experiments reflects all those who completed all of the levels.

3) Experimental Design: Our system was made available in the Internet and our experiment advertised in different mailing lists from the Universidade Federal de Viçosa, in Brazil. Participation was anonymous and volunteered.

In the beginning of the experiments the subjects were instructed about the controls of the game before playing a practice level. The practice level was the same for all participants and generated by the NLG system. Only after playing the practice level the participants evaluated the levels generated by the PCG systems. After playing each level the participants gave scores according to the criteria described above. In the end of the experiment the subjects filled a questionnaire informing their age, and their reported skills in the game of Mario (i.e., how much Mario they played before).

TABLE I: Empirical comparison of HCS-P and HCS-R. Lower values of enjoyment and visual aesthetics indicate levels that are perceived to be more enjoyable to play and have better visual aesthetics. Each entry of the table shows the average, standard deviation, and median value of the evaluations.

	HCS-P	HCS-R
Enjoyment	[2.24±1.75, 1]^a	[2.70±1.91, 2]^b
Visual Aesthetics	[2.32±1.65, 2]^a	[2.38±1.64, 2]^a
Difficulty	[3.46±1.76, 3]^a	[3.38±1.72, 3]^a

Since we noticed a strong positive correlation between enjoyment and difficulty during the annotation of the segments, in order to test the above hypothesis we needed to have the levels' difficulty as a scientific constant across all systems. This way the difficulty of the levels generated by each system would not bias the results. We manually tuned the PCG systems so that they would generate levels of similar difficulty. This was done in the HCS approaches by bounding the difficulty value used in the tension arcs to 5, and in the NLG approach by choosing a difficulty value d at random from the following options: $\{3, 4, 5\}$. We tried several different bounding values for the HCS approach and NLG, until we believed all systems were generating levels with similar difficulty. We asked the participants to report their perceived difficulty mainly to ensure that we indeed managed to control the difficulty of the levels generated by the systems tested.

Also aiming at standardizing as much as possible the conditions under which the subjects evaluated the systems, we ensured that all systems generated levels of size 160×15.

C. Results Experiment 1

The average, standard deviation, and median results of Experiment 1 are shown in Table I. The small difference in the difficulty scores is an evidence that difficulty was indeed controlled in our experiment, allowing a fair comparison of the HCS approaches. Within each row, we use different superscript letters to indicate that the results are statistically significant according to Wilcoxon signed-rank tests (Shapiro-Wilk tests show that our data is unlikely to be normally distributed: $p<0.05$ for all criteria).

Enjoyment is the only criterion in which we observed a significant difference between HCS-P and HCS-R. HCS-P

TABLE II: Empirical evaluation of PCG systems. Lower values of enjoyment and visual aesthetics indicate levels that are perceived to be more enjoyable to play and have better visual aesthetics. Each entry of the table shows the average, standard deviation, and median value of the evaluations.

	HCS-P	NLG	ORE
Enjoyment	$[2.41 \pm 1.70, 2]^a$	$[2.83 \pm 1.94, 2]^a$	$[3.56 \pm 1.91, 4]^b$
Aesthetics	$[2.41 \pm 1.57, 2]^a$	$[3.00 \pm 1.78, 3]^b$	$[3.32 \pm 1.98, 3]^b$
Difficulty	$[3.45 \pm 1.79, 3]^a$	$[3.62 \pm 2.14, 3]^a$	$[3.09 \pm 1.97, 2]^a$

generates levels which are significantly more enjoyable to play than those HCS-R generates ($p<0.05$, $r=0.36$).

1) Testing H1: In general the participants enjoyed playing the levels generated by both HCS-P and HCS-R systems. For example, a score of two and three for enjoyment means that the participant agrees and somewhat agrees, respectively, that the level is enjoyable to play. However, participants enjoyed playing levels generated by the HCS-P systems significantly more than those generated by the HCS-R system ($p<0.05$). Moreover, the difference in enjoyment between the levels generated by HCS-P and HCS-R is substantial, as the effect size is around the medium mark ($r=0.36$). These results support H1, as the people reported the levels generated by HCS-P to be more enjoyable than those generated by HCS-R.

2) Testing H2: In general the participants liked the visual aesthetics of the levels generated by both systems. Moreover, there is virtually no difference between the scores obtained by HCS-R and HCS-P: 2.42 and 2.48, respectively. This result is interesting because it suggests that visual aesthetics might be evaluated locally (i.e., by evaluating level segments). By contrast, according to the H1 results, enjoyment might have to be evaluated globally (i.e., by evaluating the entire level).

D. Results Experiment 2

The average, standard deviation, and median results for Experiment 2 are presented in Table II. Similar to Experiment 1, numbers with different letters in a given row of Table II indicate statistically significant results. Shapiro-Wilk tests showed the data obtained in Experiment 2 is unlikely to be normally distributed ($p<0.05$ for all criteria). Since now we are testing multiple systems we first run the non-parametric Friedman's test for each criterion. The results on enjoyment are statistically significant ($\chi^2(2)$=13.9, $p<0.05$) as well as on visual aesthetics ($\chi^2(2)$=6.13, $p<0.05$) across the systems; there was no statistical significance for difficulty.

We now turn to post-hoc tests (Wilcoxon signed-rank) to make pairwise comparisons of the different systems on enjoyment and visual aesthetics. In addition to the p-values, we also show the effect size r of each comparison.

There was no statistical difference in enjoyment between HCS-P and NLG ($p=0.082$). The differences between HCS-P and ORE ($p<0.05$, $r=0.53$) and between NLG and ORE ($p<0.05$, $r=0.36$) are significant. Although there was no statistical difference between HCS-P and NLG, the numbers suggest a preference for HCS-P. The difference in aesthetics between HCS-P and NLG ($p<0.05$, $r=0.36$) and between HCS-P and ORE ($p<0.05$, $r=0.33$) are significant. There was no statistical difference between NLG and ORE ($p=0.51$).

1) Testing H3: Our results partially support the hypothesis that HCS is able to generate levels which are perceived to be more enjoyable to play than the other systems tested. This is because HCS-P had the best score for enjoyment (2.41) but the difference between HCS-P and NLG is not statistically significant. Nevertheless, there is a clear trend showing a preference for the levels generated by HCS-P over those generated by NLG. Moreover, the difference between HCS-P and ORE is statistically significant and substantial as the effect size was large ($r=0.53$). The difference between NLG and ORE had a medium effect size ($r=0.36$).

2) Testing H4: Our results fully support the hypothesis that HCS-P generates levels with better visual aesthetics than NLG and ORE. The medium effect sizes shown in the comparisons between HCS-P and the other systems ($r=0.36$ for NLG and $r=0.33$ for ORE) demonstrate that the difference between HCS-P and the other systems is substantial.

E. Discussion

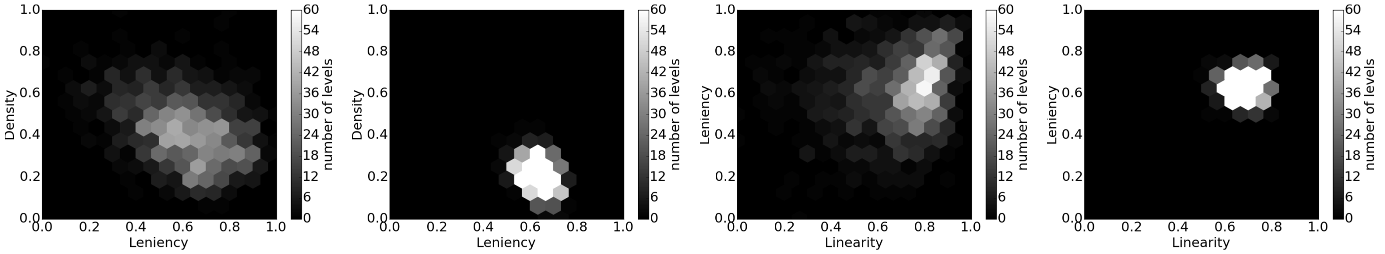
The results of Experiment 1 confirmed the conventional wisdom in level design. That is, people tend to find more enjoyable the levels whose most difficult challenges happen toward the end of the level [7]. The results of Experiment 1 also suggest that the participants' perceived visual aesthetics depends only on the state of the screen at a given time, rather than on the dynamics of their interactions (i.e., on the progression arc used to control the segments' difficulty).

The results of Experiment 2 show that HCS had better average scores for both enjoyment and visual aesthetics than NLG and ORE. The success of HCS is likely due to the synergy of its components. That is, the evaluation of the segments by human workers allows HCS to only use segments that are deemed as visually pleasing and enjoyable by people, such as the segment shown in Figure 2c. By the same argument, HCS is unlikely to use segments whose objects do not play a role in the challenges of the level, such as the segment shown in Figure 2a. Another important component of HCS is its progression arc, that implements the idea of placing the hardest challenges of a level l toward the end of l. In addition to being a common level design approach, we verified the effectiveness of this progression arc in Experiment 1.

VIII. Computational Metric Study

In this section we study how the levels generated by HCS differ from the segments in the library Γ in terms of expressive range [45], which is defined as the range of linearity and leniency values of levels generated by a given system. We use density in addition to linearity and leniency in this study.

Linearity was introduced by Smith and Whitehead [45] and is defined as the average distance between objects in the game and their predicted location according to a linear regression model. In IMB, this notion is adapted to platforms and mountain objects in the grid. Specifically, linearity measures the average distance between the center point of an object in each column in the level's grid and the predicted y coordinate



(a) segments (b) HCS-P (c) segments (d) HCS-P

Fig. 7: Expressive Range of HCS-P and of the segments composing HCS levels, which were generated by NLG.

for the object. The linearity values are first multiplied by (-1) and then normalized to a value in $[0, 1]$ such that higher values in the $[0, 1]$ range correspond to higher linearity.

Leniency approximates the difficulty of play of a level, and is computed as a weighted average of leniency terms over all objects in the level. Shaker et al. [25] suggested the following weights for IMB, which we adopt in this work: power-up items are assigned a weight of 1, cannons, flower tubes, and gaps are assigned a weight of -0.5, and enemies are assigned a weight -1. Values are normalized between $[0, 1]$ such that higher values of leniency imply a more challenging level.

Density measures the fraction of space in a level that is occupied by objects. Similar to other implementations of density (e.g., [25]), we measure the density of level l as the percentage of grid cells of l which are occupied by mountains. That is, the density value of level l is computed as $\frac{m_l}{S}$, where m_l is the number of grid cells occupied by mountains in l and S is the total number of grid cells in l. For the full-sized levels $S = 2,400$ (160×15) and for segments, $S = 300$ (20×15). Density values are also normalized to a value in $[0, 1]$ such that higher values indicate denser levels.

We use linearity, leniency, and density because we are interested in metrics that might complement our user study, which is in contrast with metrics that correlate with the scores of a user study, such as Summerville et al.'s metrics [46].

Figure 7 shows the expressive range of HCS-P (Figures 7b and 7d), and of the segments in Γ (Figures 7a and 7c). Each plot shown in Figure 7 considers exactly 1,437 levels, the size of Γ. Lighter colors indicate more levels being generated for a given pair of metric values; the number of levels for different shades of gray is shown on the right-hand side of each plot.

We observe in Figure 7 that the metric-value variability of the segments constrains the variability of the levels HCS-P generates. For example, there are few segments with leniency value as low as 0.2. As a consequence, HCS-P is unable to generate levels with such a low leniency. Similarly, there are no segments with density values as high 0.9, and that explains why HCS-P is unable to generate levels with such a high density. This analysis is important because it shows that the expressive range of HCS-P is constrained by the expressive range of its segments. Thus, for example, if one is interested in creating non-lenient HCS-P levels, then the method has to be provided with a collection of non-lenient segments.

We also observe that the variance over the metrics is larger for the set of segments than for the levels generated by HCS. For example, see Figure 7a where we observe a large number of segments with low density (around 0.3) and high leniency (around 0.9), but no HCS levels with similar density and leniency values. However, the levels generated by HCS-P are likely to include at least one segment with low density and high leniency in its initial parts. This is because HCS-P's progression arc requires a segment that is deemed as easy by human workers, and often a segment that is considered easy by humans has low density and high leniency [47]. The HCS plots do not show levels with low density and high leniency because the low density and high leniency values of the initial segments composing a level generated by HCS average out with high density and low leniency values of other segments of the level. Similar effect is observed with respect to linearity, i.e., there are many segments with large linearity values (around 0.9) and no HCS levels with such large linearity values.

IX. Conclusions

In this paper we introduced HCS, a PCG system for IMB that uses human computation in its generation process. HCS uses an existing PCG system to generate a large number of segments which are subsequently evaluated by human workers. Then, HCS uses a progression arc to combine a number of annotated segments into a full-sized level of the game. The results of a systematic user study showed that (i) the levels generated by the HCS were perceived to be more visually pleasing than the levels generated by the original PCG system as well as other PCG approaches; (ii) the levels generated by the HCS approach were perceived to be more enjoyable to play than an alternative system. Our results demonstrate the potential of the human-in-the-loop approach for PCG tasks.

X. Acknowledgements

We thank J. Mariño for implementing the metrics used in this paper, N. Tractinski and the reviewers for helpful comments. This research was supported by CAPES and FAPEMIG.

References

[1] N. Shaker, J. Togelius, and M. J. Nelson, *Procedural Content Generation in Games: A Textbook and an Overview of Current Research*. Springer, 2015.

[2] N. Shaker, G. N. Yannakakis, and J. Togelius, "Towards automatic personalized content generation for platform games," in *Conference on Artificial Intelligence and Interactive Digital Entertainment*. AAAI Press, 2010, pp. 63–68.

[3] A. M. Smith, C. Lewis, K. Hullet, and A. Sullivan, "An inclusive view of player modeling," in *Proceedings of the International Conference on the Foundations of Digital Games*. New York, NY, USA: ACM, 2011, pp. 301–303.

[4] K. Compton and M. Mateas, "Procedural level design for platform games." in *Conference on Artificial Intelligence and Interactive Digital Entertainment*. AAAI Press, 2006, pp. 109–111.

[5] W. Mason and S. Suri, "Conducting behavioral research on amazon's mechanical turk," *Behavior research methods*, vol. 44, no. 1, pp. 1–23, 2012.

[6] A. J. Quinn and B. B. Bederson, "Human computation: A survey and taxonomy of a growing field," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2011, pp. 1403–1412.

[7] C. W. Totten, *An architectural approach to level design*. CRC Press, 2014.

[8] J. Togelius, N. Shaker, S. Karakovskiy, and G. N. Yannakakis, "The Mario AI championship 2009–2012," *AI Magazine*, vol. 3, no. 34, pp. 89–92, 2013.

[9] W. M. P. Reis, L. H. S. Lelis, and Y. Gal, "Human computation for procedural content generation in platform games," in *Conference of Computational Intelligence and Games*. IEEE, 2015, pp. 99–106.

[10] G. Smith, M. Cha, and J. Whitehead, "A framework for analysis of 2d platformer levels," in *ACM SIGGRAPH Symposium on Video Games*. ACM, 2008, pp. 75–80.

[11] V. Valtchanov and J. A. Brown, "Evolving dungeon crawler levels with relative placement," in *Proceedings of the Fifth International C* Conference on Computer Science and Software Engineering*. ACM, 2012, pp. 27–35.

[12] J. Togelius, M. Preuss, N. Beume, S. Wessing, J. Hagelbäck, G. N. Yannakakis, and C. Grappiolo, "Controllable procedural map generation via multiobjective evolution," *Genetic Programming and Evolvable Machines*, vol. 14, no. 2, pp. 245–277, 2013.

[13] L. Ferreira and C. F. M. Toledo, "A search-based approach for generating angry birds levels," in *Proceedings of the Conference on Computational Intelligence and Games*, 2014, pp. 1–8.

[14] L. Cardamone, D. Loiacono, and P. L. Lanzi, "Interactive evolution for the procedural generation of tracks in a high-end racing game," in *Proceedings of the Annual Conference on Genetic and Evolutionary Computation*. ACM, 2011, pp. 395–402.

[15] M. Cook and S. Colton, "Multi-faceted evolution of simple arcade games," in *Proceedings of the Conference on Computational Intelligence and Games*. IEEE, 2011, pp. 289–296.

[16] J. R. H. Mariño and L. H. S. Lelis, "A computational model based on symmetry for generating visually pleasing maps of platform games," in *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 2016, pp. 65–71.

[17] G. Smith, J. Whitehead, and M. Mateas, "Tanagra: a mixed-initiative level design tool," in *Proceedings of the International Conference on the Foundations of Digital Games*. ACM, 2010, pp. 209–216.

[18] N. Sorenson, P. Pasquier, and S. DiPaola, "A generic approach to challenge modeling for the procedural creation of video game levels," *IEEE Transactions on Computing Intelligence and AI in Games*, vol. 3, no. 3, pp. 229–244, 2011.

[19] A. Summerville and M. Mateas, "Mystical tutor: A magic: The gathering design assistant via denoising sequence-to-sequence learning," pp. 86–92, 2016.

[20] M. Guzdial and M. O. Riedl, "Game level generation from gameplay videos," in *Proceedings of the Twelfth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 2016, pp. 44–50.

[21] N. Shaker, G. Yannakakis, and J. Togelius, "Crowdsourcing the aesthetics of platform games," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 5, no. 3, pp. 276–290, 2013.

[22] S. Snodgrass and S. Ontañón, "A hierarchical approach to generating maps using Markov chains," in *Proceedings of the Conference on Artificial Intelligence and Interactive Digital Entertainment*, 2014, pp. 59–65.

[23] J. Togelius, R. De Nardi, and S. M. Lucas, "Making racing fun through player modeling and track evolution," in *Proceedings of the SAB Workshop on Adaptive Approaches for Optimizing Player Satisfaction in Computer and Physical Games*, 2006.

[24] A. Liapis, G. N. Yannakakis, and J. Togelius, "Towards a generic method of evaluating game levels." in *Conference on Artificial Intelligence and Interactive Digital Entertainment*. AAAI Press, 2013.

[25] N. Shaker, M. Nicolau, G. N. Yannakakis, J. Togelius, and M. O'Neill, "Evolving levels for Super Mario Bros using grammatical evolution," in *Conference of Comp. Intell. and Games*. IEEE, 2012, pp. 304–311.

[26] A. Liapis, G. N. Yannakakis, and J. Togelius, "Designer modeling for sentient sketchbook," in *IEEE Conference on Computational Intelligence and Games*. IEEE, 2014, pp. 1–8.

[27] N. Shaker, M. Shaker, and J. Togelius, "Evolving playable content for cut the rope through a simulation-based approach," in *AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 2013, pp. 72–78.

[28] E. Butler, A. M. Smith, Y.-E. Liu, and Z. Popovic, "A mixed-initiative tool for designing level progressions in games," in *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM, 2013, pp. 377–386.

[29] C. R. F. G. Campos, W. O. Sá, J. M. G. Teixeira, and L. H. S. Lelis, "Mixed-initiative tool to speed up content creation in physics-based games," in *Brazilian Symposium on Games and Digital Entertainment*. SBC, 2017.

[30] A. J. Quinn and B. B. Bederson, "Human computation: a survey and taxonomy of a growing field," in *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, 2011, pp. 1403–1412.

[31] L. von Ahn, B. Maurer, C. Mcmillen, D. Abraham, and M. Blum, "re-captcha: Human-based character recognition via web security measures," *Science*, vol. 321, no. 5895, pp. 1465–1468, 2008.

[32] M. J. Raddick, G. Bracey, P. L. Gay, C. J. Lintott, P. Murray, K. Schawinski, A. S. Szalay, and J. Vandenberg, "Galaxy zoo: Exploring the motivations of citizen science volunteers," *Astronomy Education Review*, vol. 9, no. 1, p. 010103, 2010.

[33] J. V. Nickerson, *Human-Based Evolutionary Computing*. New York, NY: Springer New York, 2013, pp. 641–648.

[34] E. Kamar, S. Hacker, and E. Horvitz, "Combining human and machine intelligence in large-scale crowdsourcing," in *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 2012, pp. 467–474.

[35] A. Segal, Y. Gal, E. Kamar, E. Horvitz, A. Bowyer, and G. Miller, "Intervention strategies for increasing engagement in crowdsourcing: Platform, predictions, and experiments," in *25th International Joint Conference on Artificial Intelligence (IJCAI-16)*, 2016.

[36] P. Dai, Mausam, and D. S. Weld, "Decision-theoretic control of crowd-sourced workflows," in *Twenty-Fourth Association for the Advancement of Artificial Intelligence Conference on Artificial Intelligence*, 2010.

[37] S. Sina, A. Rosenfeld, and S. Kraus, "Generating content for scenario-based serious-games using crowdsourcing." in *The Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI-14)*, 2014, pp. 522–529.

[38] B. Li, S. Lee-Urban, G. Johnston, and M. O. Riedl, "Story generation with crowdsourced plot graphs," in *Proceedings of the AAAI Conference on Artificial Intelligence*. AAAI Press, 2013, pp. 598–604.

[39] M. Kerssemakers, J. Tuxen, J. Togelius, and G. N. Yannakakis, "A procedural procedural level generator generator," in *Proceedings of the Conference on Computational Intelligence and Games*. IEEE, 2012, pp. 335–341.

[40] T. Lavie and N. Tractinsky, "Assessing dimensions of perceived visual aesthetics of web sites," *International Journal of Human-Computer Studies*, vol. 60, no. 3, pp. 269–298, 2004.

[41] R. M. Yerkes and J. D. Dodson, "The relation of strength of stimulus to rapidity of habit formation," *Journal of Comparative Neurology and Psychology*, vol. 18, pp. 459–482, 1908.

[42] P. A. Mawhorter and M. Mateas, "Procedural level generation using occupancy-regulated extension." in *Conference of Comp. Intell. and Games*. IEEE, 2010, pp. 351–358.

[43] M. Guzdial, N. Sturtevant, and B. Li, "Deep static and dynamic level analysis: A study on infinite mario," in *Proceedings of the 3rd Experimental AI in Games Workshop*, 2016, pp. 31–38.

[44] G. N. Yannakakis and J. Hallam, "Ranking vs. preference: A comparative study of self-reporting," in *Affective Computing and Intelligent Interaction*. Springer, 2011, pp. 437–446.

[45] G. Smith and J. Whitehead, "Analyzing the expressive range of a level generator," in *Proceedings of the Workshop on Procedural Content Generation in Games*. ACM, 2010, pp. 1–7.

[46] A. Summerville, J. R. H. Mariño, S. Snodgrass, S. Ontañón, and L. H. S. Lelis, "Understanding mario: An evaluation of design metrics for platformers," in *Proceedings of the International Conference on the Foundations of Digital Games*. ACM, 2017, pp. 8:1–8:10.

[47] J. R. H. Mariño, W. M. P. Reis, and L. H. S. Lelis, "An empirical evaluation of evaluation metrics of procedurally generated Mario levels," in *AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 2015.