Article

Virtual Forestry Generation: Evaluating Models for Tree Placement in Games

Benjamin Williams [1], Panagiotis D. Ritsos [2] and Christopher Headleand [1]

[1] University of Lincoln, School of Computer Science, UK
[2] Bangor University, School of Computer Science and Electronic Engineering, UK
* Correspondence: bwilliams@lincoln.ac.uk
† This paper is an extended version of our paper published in CGVC 2019.

Version March 5, 2020 submitted to Computers

Abstract: A handful of approaches have been previously proposed to generate procedurally virtual forestry for virtual worlds and computer games, including plant growth models and point distribution methods. However, there has been no evaluation to date which assesses how effective these algorithms are at modelling real-world phenomena. In this paper we tackle this issue by evaluating three algorithms used in the generation of virtual forests – a randomly uniform point distribution method (control), a plant competition model, and an iterative random point distribution technique. Our results show that a plant competition model generated more believable content when viewed from an aerial perspective. Interestingly however, we also found that a randomly uniform point distribution method produced forestry which was rated higher in playability and photorealism, when viewed from a first-person perspective. We conclude that the objective of the game designer is important to consider when selecting an algorithm to generate forestry, as the algorithms produce forestry which is perceived differently.

Keywords: Procedural Content Generation; Virtual Forests; Computer Graphics; Video Games

---

1. Introduction

Procedural content generation is progressively becoming an established tool in the development of video games. This is especially true in the case of virtual environments and landscapes, which is particularly labour-intensive when designed by hand. Due to the advent of procedural generation, content can be automatically generated, tackling this issue by reducing development time and production costs. Furthermore, procedural generation also enables the possibility of pseudo-infinite worlds and on-the-fly content creation, amongst other things. These are just a few reasons which has driven research in this area, with approaches seeking to generate a diverse range of environmental assets. One area which particularly receives little to no attention is the prodedural creation of forest and woodland bodies. In the case of natural landscapes, vegetation is a common and important element within the virtual environment. This is especially evident in modern video games, where forestry is frequently used as part of the in-game world. Virtual forests may not only be used as scenery elements, but to enhance game mechanics for, say, providing cover to players in first-person shooter games.

The alternative to a procedural methodology is through a manual or semi-automatic design process. In the case of in-game forest scenes, this would involve the manual distribution of individual trees within the virtual world by an environment artist. However, a few problems arise when following this approach. Namely, this process is not only time-consuming, but the quality of the resulting scene is reliant on the subjective considerations of the designer. One method of circumventing these issues is by randomly sampling positions where trees are subsequently placed at. However, this approach is not representative of the way natural, real-life forests propagate. Instead, natural forests are governed by the developmental cycles of an ecosystem, spanning hundreds of years [1]. Approaches to model the distribution of these trees, should therefore, ideally, reflect this process. As a result, there has been a handful of papers which delve into ecosystem models and methods of generating plant communities, which are explored later in our discussion.

This paper extends our previous effort [2] to undertake this challenge, by introducing a handful of generation techniques and placement strategies, followed by a survey, evaluating each method in terms of perceived realism and playability. Furthermore, the attributes of the generated forestry (such as the density of the trees) are also studied to measure their impact on a player's perception of a generated forest. This has a clear application in the games development sector, as forestry is a common asset found in games, and designers need to consider which procedural approach best suits the experience they try to create for a player. With this in mind, the hypotheses for this paper are:

H1: A method which is an approximation of a real-life process (a bio-inspired approach) is perceived to generate more enjoyable and realistic content, over a stochastic method which uses randomness to distribute trees.

H2: The canopy coverage of each forest is a significant variable in the perceived playability and realism of it.

The structure of this paper is as follows: Section 2 provides an overview of procedural content generation algorithms, and a review of their use in generating virtual foliage and flora communities. Section 3 presents three different approaches in procedural forest generation and spatial distribution of trees within a virtual environment. Sections 4 and 5 discuss our pilot and main evaluations respectively, whereas section 6 presents an extended set of results. Section 7 presents a frequency analysis of the user's subjective selection counts and Section 8 concludes, also discussing future work.

2. Background

In procedural content generation, content is generated stochastically via algorithms [3,4]. This category of approaches has found success in a number of domains, including both research and commercial applications [5,6]. Interest in procedural content generation for games was born from early computer systems of the time and their inherent technical limitations [7]. Today, such approaches can be applied to synthesize a broad spectrum of virtual content, ranging from terrain height-maps [8–11], buildings and their furnishings [12,13], to the placement of assets for an entire level for a video game [14], such as settlements [15] or as in our case, plant ecosystems [16].

Procedural generation techniques have been applied specifically to the generation of simulated vegetation. The majority of existing research into procedurally generated vegetation focuses on generating individual items of vegetation, rather than an ecosystem built from individual plants. One of the most prominent methods for generating virtual trees procedurally, is through the use of Lindenmayer Systems (L-Systems) [17]. L-Systems can be used to create fractal-like patterns, using re-writable grammars [18]. These types of system are often used to generate the skeletal branches and stems of virtual trees [19–22]. In the work of Livny et al. [23], the authors even proposed an algorithm which reconstructed the skeletal system of a tree from a point-cloud through the use of L-Systems. The generation of other parts of a tree's structure, such as the bark, can also be generated procedurally. This was demonstrated by Dale et al. [24], in which the authors proposed a procedural technique for generating bark patterns, through a biomechanical physics model which emulated fractures in a tree's surface over time.

Procedural methods have also been applied to generate other forms of vegetation, such as mushrooms [25] or lichens [26]. An example of the earliest research in procedurally generating of systems of multiple plants is by Reeves and Blau [27], who explored the problem of how to generate virtual forests. A technique was developed which uses particle systems to approximate individual trees. The designer first defines a few parameters, such as the minimum distance between trees and the height-map of the terrain to place trees on. The algorithm then randomly distributes procedurally generated trees within the environment suited to the supplied parameters. Another related class of algorithms are point distribution methods. There have been a number of papers which show their use in the procedural placement of objects, including trees and forestry [28,29]. A recent example of this is by Ecormier et al. [30], in which a variance-aware disk-based distribution algorithm is presented. In particular, the authors highlight its usage in synthesising virtual forest scenes.

Other approaches, which consider plant competition models, have been developed. Plant competition models consider the simulation of each plant in an ecosystem, and interactions with its neighbours. Such an approach is presented by Bauer et al. [16] where the authors describe the *field-of-neighbourhood* (FON) model. The FON is a circular radius around each tree which determines the zone in which this tree competes with others in the community. If the FON of a tree overlaps with another tree's FON, then these trees are in competition with each other for resources. Otherwise, if there is no overlap between a tree's FON and another, then this tree is not in competition with any others. An illustration of this can be seen in Figure 1. There are two competition models to consider if the FON of two or more plants overlaps: symmetric competition and asymmetric competition. Alsweis and Deussen [31] define these as:

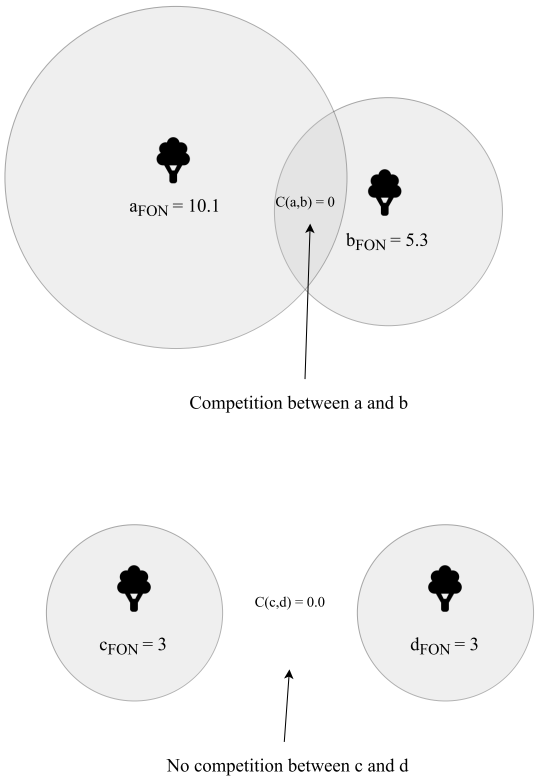


Figure 1. A diagram illustrating the *field-of-neighbourhood* (FON) model as described by Bauer et al. [16]. The top-most image shows arbitrary competition between two plants with different FON radii. The bottom-most image similarly shows two separate trees, but with no competition between them.

- **Symmetric competition:** When considering the competition between two plants, resources are split evenly between the two. This infers that the two plants are of the same size, and pose an equal threat to one another:

$$I(a,b) = \frac{C(a,b)}{2}$$

  where $C(a,b)$ yields the competition/FON-overlap between the two plants.

- **Asymmetric competition:** In the case of two plants, resources are split unevenly between the two, based on which FON is larger. This means that the tree with the smaller FON will be dominated by its competitor, resulting in no access to resources and its eventual death:

$$I(a,b) = \begin{cases} C(a,b) & \text{if } a_{\text{FON}} > b_{\text{FON}} \\ C(a,b) \text{ or } 0 & \text{if } a_{\text{FON}} = b_{\text{FON}} \\ 0 & \text{if } a_{\text{FON}} < b_{\text{FON}} \end{cases}$$

Alsweis and Deussen [31] use bio-inspired rules coupled with the FON model to generate plant communities through asymmetric competition. The development of a plant depends on a designer-supplied map which represents the amount of nutrition found throughout the terrain. Members of the simulated plant community reproduce by spreading their seed locally once they reach a certain size. The seed production of each tree also grows alongside its size – as it increases in size, it produces more seeds as a result. A 'mortality risk' is also introduced into the system, in which plants which fall below the average plant size are culled due to competition. Computer applications such as GREENLAB [32] have also been developed to generate and study various bio-inspired growth models. Cournede et al. [33] used this application to study forest growth and propose a software system to compose virtual forest scenes. Lane and Prusinkiewicz [34] use a similar approach to develop plant communities. In their method, a plant community is represented as a multiset L-System, in which individual strings of the L-System represent a tree. This multiset of strings is then added to or removed from to simulate growth within the forest. The authors also describe similar concepts, such as a radius around each tree in which it interacts with others (similar to the FON model) and domination of resources through asymmetric competition. To do this, the authors introduce the following three steps for each tree in the multiset:

- **Self-thinning:** A similar notion to asymmetric competition – plants which are in competition with larger ones are dominated, and are subsequently culled from the population. Competition is also detected in a similar method to the FON model[31]. That is, if the radii of two trees overlap, the two plants are in competition with one another.
- **Succession:** Trees grow over time, and have a random probability of dying at each step once they reach a certain age. This ensures that old trees are culled from the population.
- **Plant propagation:** Trees reproduce in a similar method proposed by Alsweis and Deussen [31], in which seeds are sown locally around the tree chosen for reproduction. This helps to cluster trees together which are of the same species.

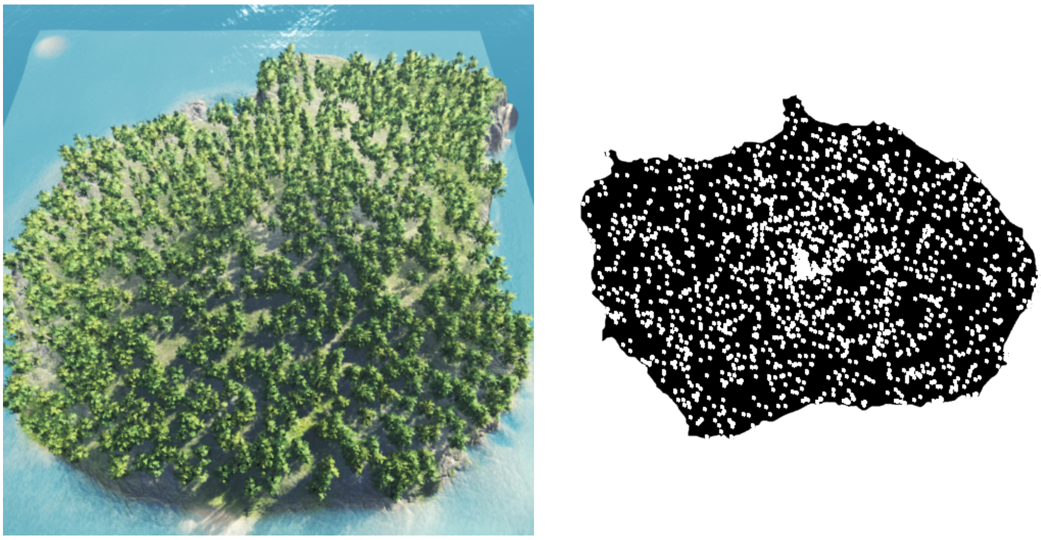
Cordonnier et al. [9] draw attention to some scalability issues of FON-based competition models. In particular, the computational expense of FON models is moderate in smaller-scale simulations, but infeasible at larger scales. The authors introduce an approach to procedurally generate ecosystems with combined terrain generation. Instead of using a FON-based model, a non-competitive cell-based approach is used to simulate growth. In this approach, the landscape is subdivided into cells, and ecosystem events are generated at random in a given cell. Plant growth, death and germination are simulated based on plant viability. Plant viability is calculated by taking into account local temperature, soil moisture and sun exposure, amongst other factors.

3. Forest Generation Approaches

In this section we introduce three algorithms for the spatial distribution of trees within an environment. The first, the *Naive* algorithm, is provided as a baseline to evaluate the other methods against. This algorithm uniformly distributes trees randomly within the environment and is commonly used in games development. The second method is *Propagation*, based on a asymmetric plant competition technique, which implements the FON model discussed previously. This algorithm is a bio-inspired approach intended to approximate how natural forests grow over time. The third algorithm, the *Clustering* method is provided as an intermediary between the *Naive* and *Propagation* algorithms by using an iterative random distribution technique. We have selected these three algorithms to examine the differences between plant competition models and methods which randomly sample from a distribution.

3.1. Method 1: Naive

The *Naive* method randomly distributes trees within a given area. The algorithm distributes trees by sampling a random (x, y) point in a uniform distribution, and places a tree at the sampled point. The algorithm used throughout this paper was adapted slightly to create forests at various densities. Instead of specifying a number of trees to spawn initially, a target density was specified and the algorithm ran until this target density was matched. Of all the methods described throughout this paper, the Naive method requires the least computational resources due to its simplicity.



(a) **(b)**

Figure 2. *a)* An example of a top-down virtual forest generated with the Naive algorithm, implemented in Unity 3D. b) An example in 2D.

The algorithms used in our studies are modified slightly to create forests at various densities. Instead of specifying a number of trees to spawn initially, a target density for the virtual forest is specified instead, and the algorithm is followed until this target density is matched. For example, the density d for the virtual forest in Figure 2a is $d \approx$ 60.08%), which is measured as the percentage of canopy cover across the island.

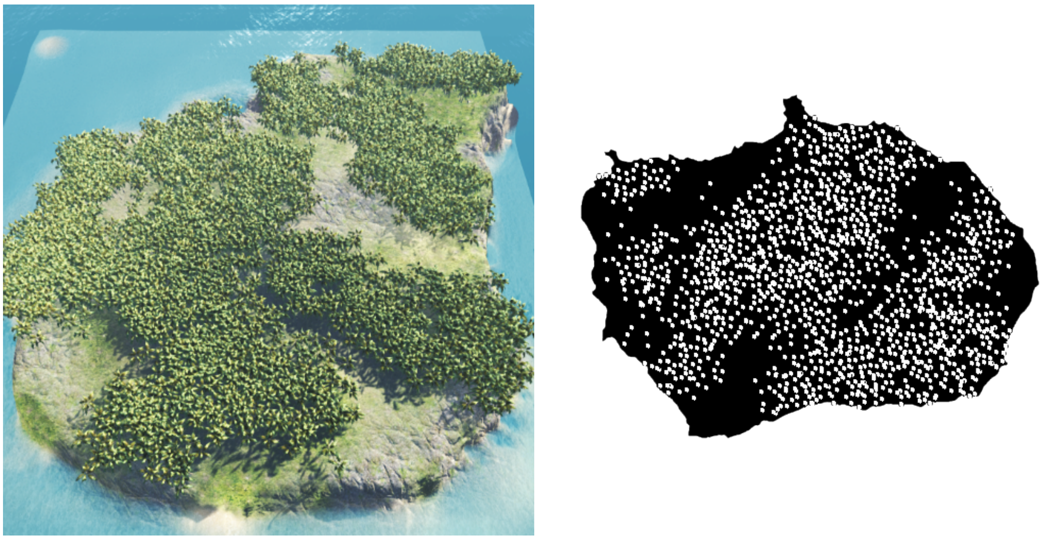
3.2. Method 2: Propagation

The *Propagation* method takes its inspiration from the rules that govern how forests develop in nature. This method should not be considered a faithful reflection of a natural process, but rather a bio-inspired approximation. To do this, this method is based on the asymmetric plant competition approach described by Lane and Prusinkiewicz [34]. We also similarly make use of a FON-based approach to represent competition between trees. Furthermore, the three steps introduced by Lane and Prusinkiewicz within our algorithm are applied:

- **Succession**: In each simulation iteration, every tree ages (and grows) until it reaches a mature age. Once a tree reaches a certain age, it dies and is culled from the population.
- **Plant propagation**: Once trees have reached a mature age, they can reproduce by sowing seeds locally to their position.

- **Self-thinning**: If a tree is growing close to another tree, then the oldest (and largest) tree will outgrow the other, thereby killing it and culling it from the environment. This is an approximation of asymmetric plant competition.

In addition to these rules, the wind direction and wind magnitude are also simulated whilst generating the virtual forest. It is important to note that this is not an accurate simulation of nature, and various factors (such as evolutionary forces) are ignored. We accept this, and have simply taken inspiration from biology to try and generate something which is visually appropriate.



(a) **(b)**

Figure 3. *a)* An example of a top-down forest image created using the Propagation algorithm, in a 3D environment. *b)* Another image generated using the same algorithm, but in a 2D environment. Both *a)* and *b)* were generated over a total of 13 iterations.

This method has the advantage of spacing trees in a fairly regular manner, which can be seen in Figure 3a and 3b. Due to the nature of the approach, trees should remain equidistant, as competition results in the smaller tree's death. However, this approach is generally more computationally expensive than point distribution methods, as it requires successive iterations and significantly more computation. This may be an issue for devices with limited computational power, such as mobile devices.

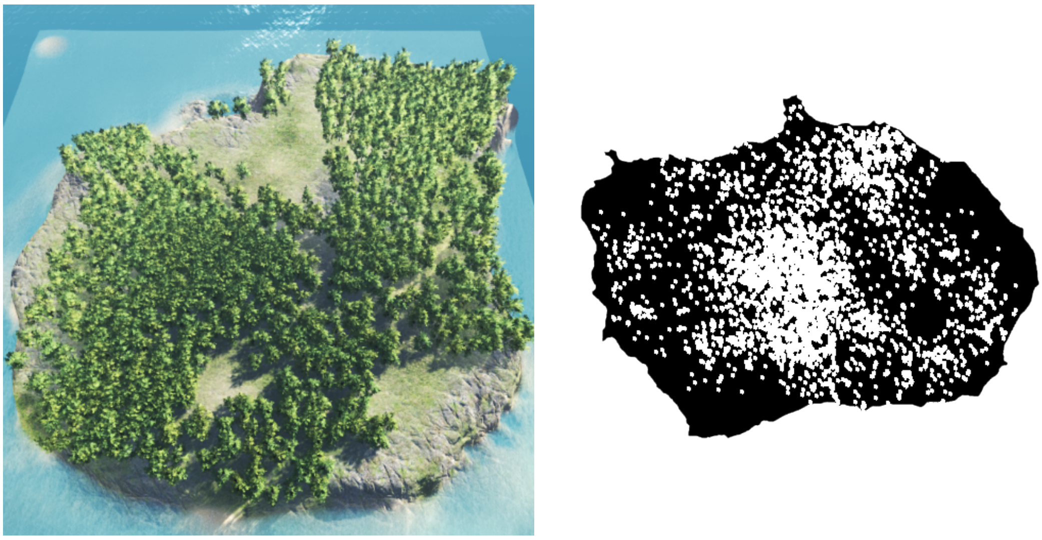
3.3. Method 3: Clustering

The *Clustering* method is an iterative random point distribution algorithm, with the goal of creating clustered areas of trees. To do this, the Clustering method initially selects a handful of random positions within the map in the first iteration, which we refer to as 'spawn points'. These are chosen in a similar fashion to the Naive approach, sampling from a uniform distribution. In the second and final iteration, points are randomly chosen within a predefined radius of each spawn point to produce clusters of trees. Tree meshes are then placed in each of these final points to produce a forest.

Likewise to the *Naive* method, the *Clustering* approach has the advantage of requiring very minimal resources, as the environment is not continuously updated and rules are not considered for each iteration of the forest's lifetime. This algorithm produces clustered distributions of trees, rather than an even and uniform distribution. Figures 4a and 4b show two examples of virtual forests generated with this algorithm, from an aerial perspective.

4. First Study: 2D Evaluation

An initial study was undertaken to evaluate whether the more complex approaches are preferred by players. The study consisted of an online survey where participants ranked images of aerial 2D



(a) **(b)**

Figure 4. *a)* An example of top-down virtual forest generated with the Clustering algorithm, in a 3D environment. *b)* A similar forest generated with the same algorithm, but in a 2D environment.

representations of forests. The objective of this evaluation was to collect preference data regarding the visual forest representations. For each question in the survey, participants were presented with three images of forests generated by each algorithm. Each image was randomly ordered on the screen, to reduce any selection bias between questions. The participant was then required to select one of these images which best matched the question criteria. The questions presented to each user throughout the survey evaluated two types of criteria. The first question was focused on the perceived realism of the environments. For these questions, the participant was asked to select two images (of the same three images) which they perceive to be the most and least realistic. The second criteria focused on the perceived suitability of the forest as an in-game environment. For this criteria, the participant were asked to imagine which environment they would (not) choose if they were to play a game based within this environment. Both of these metrics are subjective to the observer. The first relies on them comparing the image to their perception/experience of what a forest should look like. The second by comparison explores their game-play preferences, assessing whether the environments perceived to be more (or less) believable are considered more (or less) interesting to play games within.

Each participant was presented with five questions for each criteria, yielding a total of 20 individual questions. For each of the five questions, three new images were selected and presented to the participant.

4.1. 2D Study Results

The online survey was completed by 86 participants. Of these participants, 53.48% self-identified as female, with the remaining 46.52% as male. Furthermore, we also captured the general location of each participant, as the demographic featured participants from around the world.

The first and most compelling result found is the performance of the Naive distribution algorithm, which was comparatively rated higher than its competitors in terms of its perceived playability (see Figure 5). The Clustering method by comparison was rated as the method which produced the most forests perceived as most realistic. Figure 5 demonstrates that the Propagation distribution method was rated the lowest in terms of realism, but produced forests which were similar to the Clustering method in terms of playability. This same trend can also be seen for the questions which asked for the most unrealistic and unplayable environments (see Figure 6). For this category of questions, the Naive algorithm was similarly voted as the algorithm which produced the perceivably most realistic and

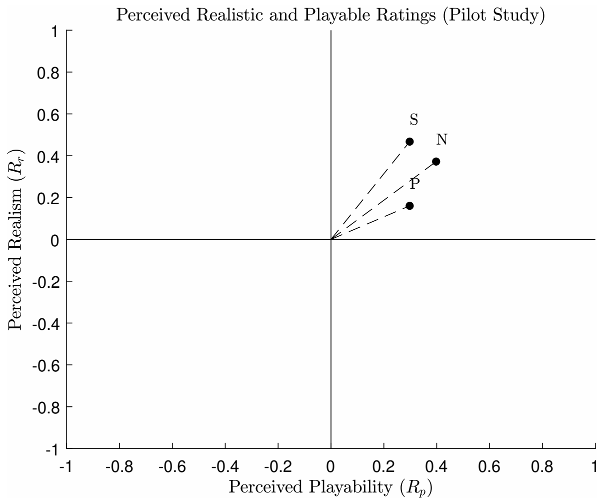


Figure 5. The normalized number of responses from participants when asked to choose the most realistic and playable forest. The letters in this figure correspond to each algorithm used.

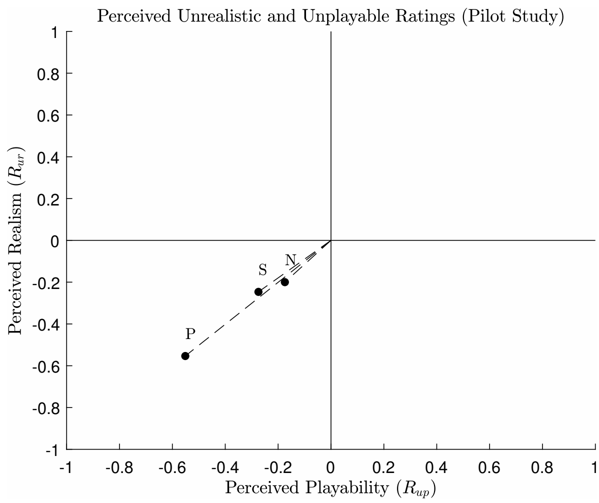


Figure 6. The normalized number of responses from participants when asked to choose the most unrealistic and unplayable forest. The letters in this figure correspond to each algorithm used.

playable environments. The Propagation algorithm however was rated as the most unrealistic and unplayable forest by a considerable margin.

Lastly, the number of ratings for each algorithm were used to provide a metric of performance, to show the overall quality of each algorithm. The metric used is calculated as $P_r = (R_r - R_{ur})$ and $P_i = (R_p - R_{up})$. R_r is the number of realistic ratings it received, R_{ur} is the number of unrealistic ratings, R_p is the number of playable ratings received and R_{up} is the number of unplayable ratings.

Figure 7 shows these two metrics plotted against each other, showing the overall performance of each algorithm. Interestingly, the performance of the Propagation algorithm was the poorest, producing the most unrealistic and unplayable environments. In contrast to this, the Clustering algorithm produced the most realistic environments, and the Naive algorithm yielded the most playable environments. It was hypothesised that the application of the Propagation algorithm would produce more realistic and playable environments, over the other two methods. However, the results show that the non-deterministic algorithms are rated higher in both categories. A further study is required to examine if this is the case under different conditions, and whether or not certain variables (such as forest density) yield similar results.

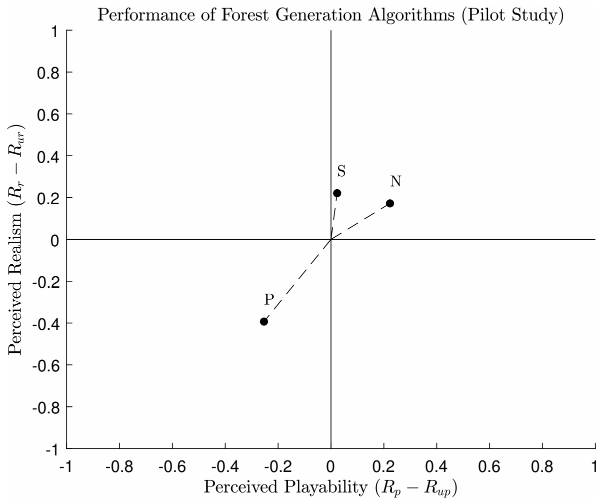


Figure 7. The overall performance of each algorithm. Here the metrics used are the difference between positive and negative ratings.

5. Second Study: 3D and Density Evaluation

A second study was conducted, in order to explore some of the questions raised by the first and to provide a more in-depth analysis of the reasoning behind selections. In this study the density of each virtual forest, along with the algorithm that produces it, were recorded and analysed. The participant also had the option of providing written feedback at every stage of each question.

As with the previous study, for each question asked, the survey presented the participant with three images to choose from. The participant would then choose the image which best suited the question that was asked. The questions were tailored in such a way to investigate whether the density or algorithm used in virtual forest propagation resulted in more playable or realistic selections. When selecting images to present to the participant, two independent variables were considered.

5.1. Algorithm Chosen

For these questions, the process started by first randomly selecting a forest density from the list of available options (Low, Medium or High). This density was then used to select three images for the participant, each of which was generated with a corresponding algorithm. For example if the randomly chosen density was 'Low', three low density forest images would be selected – one generated with the *Naive* algorithm, one with the *Clustering* algorithm, and another with the *Propagation* algorithm.

5.2. Forest Density

If the independent variable was forest density, then a similar process was followed, but showing varying forest densities generated with a single algorithm. To elaborate, an algorithm from the list of available options is randomly chosen (Naive, Clustering or Propagation). If for example, the randomly chosen algorithm was 'Naive', then three forest images generated by the Naive algorithm would be displayed to the user – one with a low density, another with a medium density, and another with a high density.

Once the three images were selected using these processes, the participant was then asked four questions about the selected images. These questions involved rating the forest images which best suited the question that was asked. These four questions were:

- 'Based on these images, which is the most realistic forest?'
- 'Based on these images, which is the least realistic forest?'
- 'If you were to play a game in one of these forests, which environment would you select to play within based on these top-down images?'

- 'If you were to play a game in one of these forests, which environment would you not select to play within based on these top-down images?'

5.3. Image Perspectives

Another limitation of the first study was that the images presented to each participant were from a single, top-down 2D perspective. This was addressed in the second study by introducing images which were rendered in 3D from two perspectives. Additionally, these images allowed further analyse if player perspective had an effect on a participant ratings. The first was a top-down perspective similar to the images from the pilot study, but rendered photo-realistically in 3D. The second used a first-person perspective situated within the forest. An example of the perspectives used in images can be seen in Figure 8. These perspectives were also used in the question selection process. The same

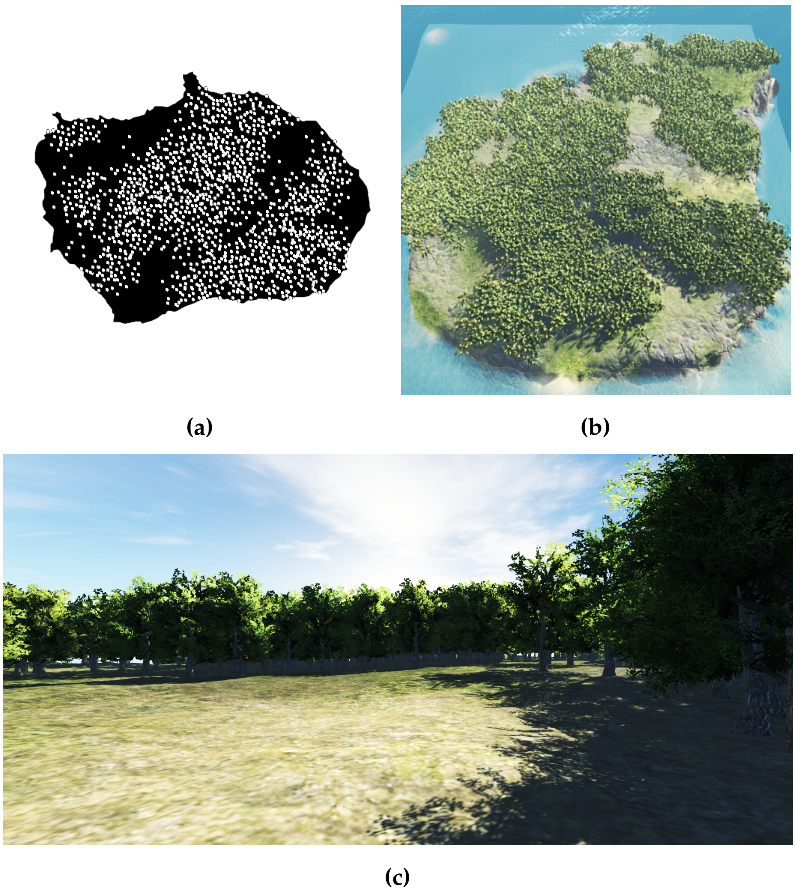


Figure 8. *a)* An example of a top-down 2D perspective, *b)* An example of a top-down 3D perspective, *c)* An example of a first-person 3D perspective.

processes outlined earlier involving the isolation of forest density and the generation algorithm were used, but for every perspective. This means that eight questions were asked for each perspective, resulting in a total of 24 questions for the participant to complete. The study ran for three weeks in total, with 71 respondents. Of these 71 respondents, 77.46% were Male, 19.71% were Female, and 2.81% did not specify their gender. The following sections analyse responses given for each perspective.

6. Results

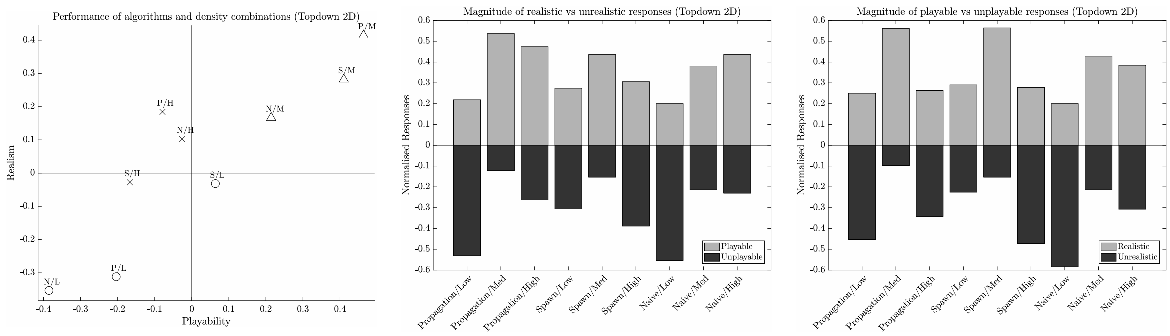
6.1. Top-down 2D Perspective

We plotted participant responses (Figure 9a), which measured the percentage a particular algorithm/density pairing (images generated with that density and algorithm) was chosen as playable versus the number of times is was chosen as realistic. The results show that images generated with the Propagation algorithm using a medium density scored higher in terms of both realism and perceived playability. An interesting result here is that the images generated with a medium density were rated similarly, and performed well in terms of both playability and realism. From this we can draw the conclusion that the most enjoyable forests for a top-down 2D perspective are generated with a medium density. It is also interesting to note that images of forests generated with a low density generally received a poor score. The exception however, are images generated with the Clustering algorithm using a low density, which was actually ranked higher in both realism and playability. Forests generated with a high density mostly scored well in terms of realism, but were rated low in terms of playability. Figures 9b and 9c show the amount of responses provided for each particular combination of algorithm and density used to generate imagery. These figures also show in general, how many times a combination was rated negatively or positively. An interesting phenomenon regarding these is the amount of negative votes, which outweigh the number of positive ones. This means that participants who rated images generated with this perspective were more prone to select a negative rating rather than a positive one.

6.2. Top-Down 3D Perspective

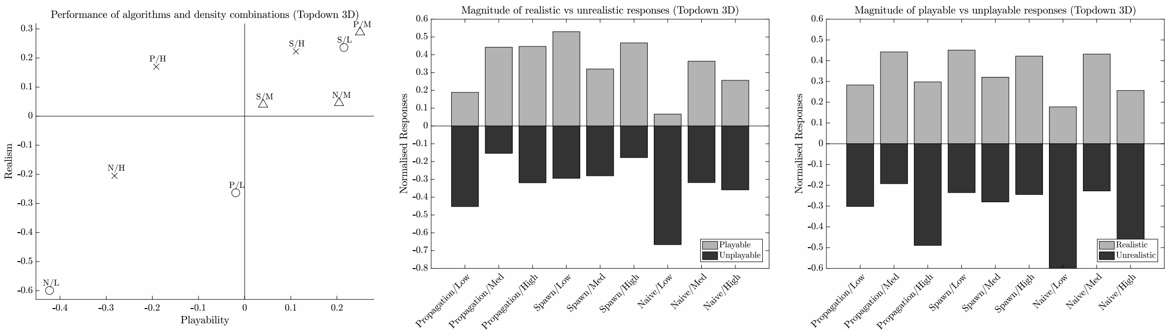
Through examination of Figure 10a, it can be seen that the results are similar to the ones found for the top-down 2D perspective (Figure 9a). Most notably, images generated with the Propagation algorithm using a medium density were again rated as the most realistic and playable environments. An interesting note however, is that images created using the Clustering algorithm have generally increased in both metrics, and are in fact some of the best performing results. Figures 10b and 10c show the number of negative and positive ratings for generated images. These results are similar to the Top-down 2D perspective.

Images generated with the Propagation algorithm with a high density were rated well in terms of realism, but poorly in terms of playability. When compared to a lower density using the same algorithm, some intriguing results were found. Images generated with the Propagation algorithm but using a low density were rated high for playability, and low in realism - the opposite of the ratings when using a high density. The same algorithm is used to generate both types of images. The only difference between these two is the change in forest density. This contrast in terms of ratings leads us to believe that there may be a correlation between forest density and the perceived playability of an environment, when using this type of algorithm to generate an image of a virtual forest.



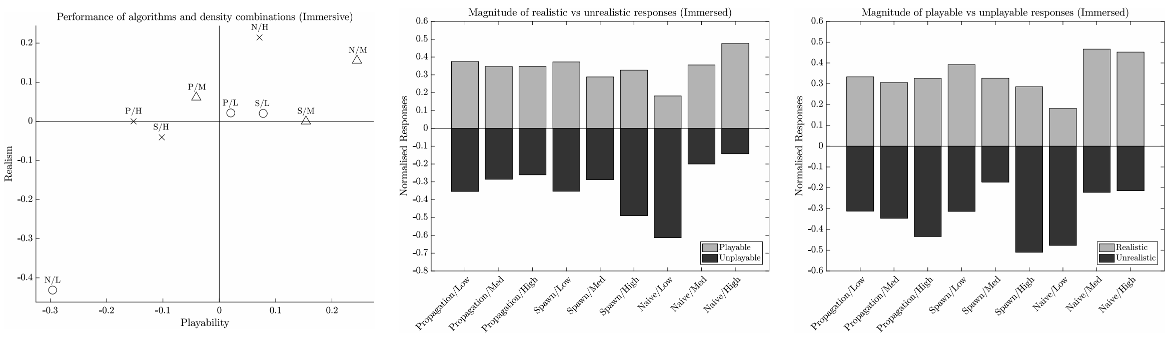
(a) (b) (c)

Figure 9. *a)* Overall performance of all algorithm and densities for top-down 2D images, realistic rating vs playability rating, *b)* Magnitude of ratings for realistic/unrealistic responses and *c)* Magnitude of ratings for playable/unplayable responses.



(a) **(b)** **(c)**

Figure 10. *a)* Overall performance of all algorithm and densities for top-down 3D images, realistic rating vs playability rating, *b)* Magnitude of ratings for realistic/unrealistic responses and *c)* Magnitude of ratings for playable/unplayable responses.



(a) **(b)** **(c)**

Figure 11. *a)* Overall performance of all algorithm and densities for first-person images, realistic rating vs playability rating, *b)* Magnitude of ratings for realistic/unrealistic responses and *c)* Magnitude of ratings for playable/unplayable responses.

6.3. First-person 3D Perspective

The results were collated in the same manner as the previous sections. Figure 11a depicts rated realism and playability of images generated with each combination of algorithm and density. Interestingly, the results in this case differ from the results for the two other perspectives. The most compelling of these differences is that images generated using the Naive algorithm with either a medium or high density were rated the most realistic and playable environments. However, images generated with the Naive algorithm and a low density were rated lowest in terms of realism and playability. Comparing the results of using the Naive algorithm with medium and high densities further confirms the same correlation discovered in the previous section, in which the density used in the generation process affected its rated playability. In this case, the same relationship is shown – a higher density is rated as less playable than a medium density. This can also be seen in the same plot with the Propagation and Clustering algorithms, where a high density is rated less playable than a medium or low density. Furthermore, these results suggest that using a pseudo-random distribution strategy results in a more playable and realistic environment for players, at least, when viewing it from a first-person perspective. This has advantages over other methods, as it is computationally inexpensive in comparison, yet yields the most believable and playable environments for this perspective. Figures 11b and 11c show the number of negative and positive votes for images generated with each combination of algorithm and forest density.

7. Frequency Analysis of Selection Counts

As mentioned previously, participants could rate images in two criteria: believability and playability. At each stage, participants are asked to choose which image satisfies them in the most and least of these criteria. This gives four possible ratings of images: the most/least believable, and the most/least playable. There are also two variables which influence the generated forest, namely the type of algorithm and the forest's density. Given these two variables, and the possible ratings each image can receive, an interesting question arises regarding the distribution of votes for images presented to participants. Observing frequency distributions will allow for interesting conclusions to be drawn from the data, for example, potential relationships between forest density and the number of times it was selected as the most believable. To achieve this, several contingency tables were created, showing the frequency of selection between different variables. These are each presented and discussed in the following sections.

7.1. Forest Density and Believability

The first area which was considered was the cross-tabulation of forest density types (low, medium and high) with other variables, which could highlight some interesting relationships. The first of these is the perceived realism of images. In particular, the frequency each density was voted by participants as the most or least believable choice. Cross-tabulations are labelled by image perspective, to explore how this variable impacted the scores given by participants. It is also worth noting that these selections were mutually exclusive, disallowing the same image to be selected for both questions.

Table 1. A table showing the number of times each type of forest density was selected as most or least believable. Notice that columns are categorised by image perspective for clarity. The labels +B and -B respectively correspond to the count of most and least believable selections. In contrast, the label U (Unrated) represents the number of times it was not selected as either.

	First-person			**Aerial (2D)**			**Aerial (3D)**		
	+B	-B	U	+B	-B	U	+B	-B	U
Low	22	37	12	4	48	19	14	41	16
Medium	23	13	35	38	2	31	31	11	29
High	26	20	25	29	17	25	26	16	29

Table 1 presents the number of times each image density was selected as most or least believable, for each image perspective. Across all three image perspectives, it can be seen that lower density forests are frequently rated as the least believable selection. This is also true across all densities, with low densities ranking the lowest in terms of believability from a first-person perspective (χ^2 (2) = 13.38, $p = 0.001$), a 2D aerial perspective (χ^2 (2) = 42.28, $p < 0.0001$) and a 3D aerial perspective (χ^2 (2) = 19.12, $p < 0.0001$). These results seem to suggest that low density distributions are generally unsuitable for generating forests which are similar to real-life, regardless of the user's viewpoint. Interestingly, the opposite effect can be seen in the case of medium densities, with medium densities being consistently selected as the most believable forest. The distribution of tallies suggests this is the case is also regardless of image perspective, whether it be first-person (χ^2 (2) = 10.25, $p = 0.006$), 2D aerial (χ^2 (2) = 30.78, $p < 0.0001$) or 3D aerial (χ^2 (2) = 10.25352, $p = 0.005$).

Of the three image perspectives, the 2D aerial perspective shows the most polarised distribution of positive/negative rating. What is particularly interesting is the differences in perceived realism across the three forest densities. For this perspective, low density forests received a particularly high number of votes as the least believable density. Conversely, both medium and high densities were chosen more frequently as the most believable. However, medium densities were substantially more polarised. It should also be noted that the same pattern of polarisation with regards to medium and high densities can be seen across all three image perspectives. This could signify that participants could more easily determine the believability of medium and low densities, in contrast to high densities.

Another interesting area is the comparison of ratings between the two aerial perspectives. Comparing both aerial perspectives reveals some interesting results. The most noteworthy difference between the two perspectives is the contrast between negative/positive ratings. In this case, 2D aerial perspectives are more polarised with respect to positive/negative selection, suggesting that image dimensionality could impact perceived believability. Curiously, this is not true of high density forests, with little to no difference in selection frequency between 2D and 3D perspectives. However, it is worth noting that a more rigorous investigation is required to conclude if this is the case.

In a similar spirit, how first-person and aerial perspectives differ in selection frequency is another area of consideration. Naturally, it could be assumed that first-person and aerial perspectives receive considerably distinct believability ratings, due to differences in how clearly the distribution of trees can be viewed as a whole. For instance, participants may find it harder to survey distributions wholly from a first-person perspective, due to the lack of a vantage point. A comparison of first-person and aerial perspectives can be seen in Table 1, highlighting a pattern of votes between the two. For example, low densities are considered significantly less believable across both first-person and aerial perspectives. Similarly, medium and high densities are considered more believable when comparing the two types of perspective. However, there is a substantially less polarisation between positive/negative votes in the case of the first-person perspective. This potentially indicates that judgement of believability may be more difficult from a first-person perspective, due to the inability to survey the distribution as a whole. Further work would be required to ascertain if this is the case, however.

7.2. Forest Density and Playability

In the preceding discussion, forest densities were cross-tabulated with believability to investigate the relationship between the two. However, believability is only one of two criteria in which participants were asked to rate images, the other being playability. Whilst believability is an interesting criteria to examine, how suitable a forest is as an environment in a video game is another important factor. For instance, exploring how the density of a generated forest affects its playability could inform level design in commercial games development. With this goal in mind, a cross-tabulation similar to the previous section was created to investigate relationships between forest density and playability. This is reflected in Table 2, which displays the frequency each density was selected as the most/least playable choice.

Table 2. A table showing the number of times each type of forest density was selected as most or least playable. Notice that columns are categorised by image perspective for clarity. The labels +P and -P respectively correspond to the count of most and least playable selections. In contrast, the label U (Unrated) represents the number of times it was not selected as either.

	First-person			**Aerial (2D)**			**Aerial (3D)**		
	+P	-P	U	+P	-P	U	+P	-P	U
Low	21	28	22	7	41	23	19	30	22
Medium	27	11	33	46	2	23	33	8	30
High	23	31	17	18	27	26	19	32	20

Perhaps the most noteworthy result is that medium density forests were consistently rated as significantly playable environments, across each of the first-person (χ^2 (2) = 10.92, $p = 0.004$), 2D aerial (χ^2 (2)= 40.92, p < 0.0005) and 3D aerial (χ^2 (2) = 15.75, $p = 0.0004$) perspectives. A similar finding was unearthed in the previous section, revealing that medium forest densities were typically selected as the most believable environments. Compounded with this result, it can be concluded that medium densities were selected most frequently in terms of both believability and playability, regardless of image perspective. One similarity between Tables 1 & 2 is that in both, the 2D aerial perspective shows the most polarised results. This suggests that participants could most easily determine both believability and playability from this perspective. Whilst this is an unexpected and interesting result, we leave the task of exploring this area to future work.

Another interesting discussion is the differences in playability votes between 2D and 3D aerial perspectives. Generally, the distribution of votes share several similarities between the two perspectives. For instance, in each case both low and high densities are rated more times as the least playable environment than the most playable. An interesting observation is the fact that high densities received more unplayable ratings than playable, with this being the case across all three image perspectives. The fact that high densities are rated so differently in believability and playability could possibly indicate a negative relationship between the two. That is, high density point distributions create believable but unplayable environments. It may be the case for example, that high density forests exhibit low tree interspacing which is considered believable, but does not result in a navigable game level. This may be a fascinating avenue of research for future work. It is worth noting however, that statistical analysis indicates the results for high densities may be subject to noise; across first-person (χ^2 (2) = 4.17, $p = 0.12$), 2D aerial (χ^2 (2) = 2.06, $p = 0.36$) and 3D aerial (χ^2 (2) = 4.42, $p = 0.11$) perspectives.

The comparison of the first person perspective against the two aerial perspectives reveals similar findings to the believability cross-tabulation. More specifically, the distribution of the most/least playable selections across all three densities follows a common pattern. In each case, low and high densities were chosen more frequently as the least playable environment. Similarly, medium densities were selected as the most playable environment. However, there is a considerable difference in polarity of negative/positive votes between first-person and aerial perspectives. In particular, the difference in negative/positive selection frequency are less extreme in the case of the first-person perspective. This is a very similar finding to the previous section, which concerned believability. Furthermore, this implies that participants found it harder to judge both believability and playability from first-person perspectives. As mentioned earlier, the lack of a vantage point could be the issue. However, further investigation would be required to identify if this is the case.

7.3. Generation Algorithm and Believability

So far, the impact of forest density on participant preferences has been discussed. Whilst the effects of forest density is an interesting area to explore, another factor in our study was the type of procedural algorithm used to generate virtual forests. Identifying how each of the three algorithms affects perceived believability/playability could give insights into which is the most preferred by players. More importantly, this could be crucial to commercial games development, whose aim is to create immersive and playable virtual environments for players. To achieve this, a similar methodology is used to the previous sections. As mentioned earlier, there were three procedural algorithms used to generate forest images. There were the naive, clustering and propagation algorithms. A cross-tabulation of generation algorithm and believability ratings can be seen below in Table 3.

Table 3. A table showing the number of times each type of generation algorithm was selected as most or least believable. The labels +B and -B respectively correspond to the count of most and least believable selections. The label U (Unrated) represents the number of times it was not selected as either.

	First-person			**Aerial (2D)**			**Aerial (3D)**		
	+P	-P	U	+P	-P	U	+P	-P	U
Naive	28	18	25	28	22	21	24	21	26
Clustering	19	31	21	24	18	29	35	11	25
Propagation	24	21	26	19	29	23	12	38	21

The first noticeable result is that the naive algorithm generally received more votes in favour of it being the most believable image, rather than the least believable. This is also the case across all three image perspectives, which could signify that participants found the naive method to be a method of creating realistic forest distributions. However, this may not be the case, as statistical analysis shows insignificant results across first-person (χ^2 (2) = 2.225, $p = 0.32$), 2D aerial (χ^2 (2) = 1.21, $p = 0.54$) and 3D aerial (χ^2 (2) = 0.535, p = 0.76) perspectives. Similarly, the clustering algorithm was rated as more believable for the 2D and 3D aerial perspectives, suggesting potential differences between a first-person and aerial perspective. Although the effect observed for the 2D aerial perspective is likely due to noise (χ^2 (2) = 2.56, $p = 0.277$), there is a significant probability that the 3D aerial perspective is not (χ^2 (2) = 12.28, $p = 0.02$). This is evidence that the clustering algorithm is a feasible alternative to procedurally generating believable tree distributions, from an aerial perspective. This may have potential impacts on games development, especially given that the clustering algorithm provides a more efficient and suitable alternative to plant growth models.

The opposite can be found for the propagation algorithm, with generated images rated significantly as the least believable, for the 3D aerial perspective (χ^2 (2) = 14.73, $p < 0.005$). The same effect is observed for the 2D aerial perspective, but lacks statistical significance (χ^2 (2) = 2.14, $p = 0.34$). Interestingly, the same cannot be said for the first-person perspective, in which the propagation algorithm received more favourable ratings than unfavourable. However, there is a considerable chance this may be due to noise too (χ^2 (2) = 0.535, $p = 0.76$). These findings suggest that generally, the propagation algorithm generates forest distributions which participants deem unbelievable from a 3D perspective. Furthermore, there are some noteworthy results when compared to the previous section, which explored the relationship of forest density and believability. Firstly, the density cross-tabulation featured boldly contrasting results with considerable polarisation between positive/negative selection counts. Furthermore, statistical tests highlighted a number of significant results and relationships. By comparison, cross-tabulating the type of procedural algorithm and selections made by participants reveals very few significant results. One explanation could be that participants find forest density a more distinguishable characteristic in assessing the believability of forest images.

7.4. Generation Algorithm and Playability

Whilst in the previous section the effects on believability were explored, another interesting and related area is how measures of playability are affected by the three algorithms used. Determining this may support games developers to create fun and challenging games, by displaying the most preferred algorithm for creating playable environments. A cross-tabulation of generation algorithm and received playability ratings can be seen in Table 4.

Table 4. A table showing the number of times each type of generation algorithm was selected as most or least playable. The labels +P and -P respectively correspond to the count of most and least believable selections. In contrast, the label U (Unrated) represents the number of times it was not selected as either.

	First-person			**Aerial (2D)**			**Aerial (3D)**		
	+P	-P	U	+P	-P	U	+P	-P	U
Naive	21	28	22	24	21	26	24	22	25
Clustering	22	23	26	25	15	31	34	11	26
Propagation	29	19	24	22	34	15	13	37	21

For the first-person perspective, there are a few contrasting results between the three algorithms. Firstly, forests generated by the Naive algorithm were selected most often as the least playable, of the three algorithms in this perspective. Conversely, forests generated by the Propagation algorithm received the highest number of most playable votes. By the same token, the number of most/least playable selections for the Clustering algorithm are practically identical. These results potentially suggest that plant growth models are the most suitable for creating playable environments from a first-person perspective. On the other hand, uniform point distribution appears to yield the least playable environments in this perspective. Interestingly, almost the opposite effect can be seen from the 2D aerial perspective. Most noticeably, the Propagation algorithm was rated significantly as the algorithm which produces the least playable environments (χ^2 (2) = 7.802, $p = 0.02$). To contrast, the Clustering algorithm was preferred in creating the most playable environments of the three algorithms, with this perspective in mind. The clear difference in selections between these the first-person and aerial perspectives shows that image perspective is a considerable part of how forests are judged in perceived playability. The 3D aerial perspective also shared a few commonalities to the 2D aerial perspective. For example, the Naive algorithm was rated almost identically to the 2D aerial perspective. Furthermore, the Clustering algorithm was considered the most frequently as creating the most playable environments (χ^2 (2) = 11.52, $p = 0.003$), and the Propagation algorithm as the least (χ^2 (2) = 12.61, $p = 0.001$).

There is also a considerable difference between selection counts in the first-person and aerial perspectives. Whilst the two aerial perspectives share different selection counts, they are very similar in nature. Perhaps the most glaring result is the selection frequency of the Propagation algorithm, which is generally rated well from the first-person perspective, but negatively in the two aerial perspectives. Further research would be required to ascertain why this is the case.

7.5. Summary

An in-depth look at forest selection counts has unearthed some results worthy of discussion. The focus of our analysis was to understand how participants perceive generated forests, for different sets of generation parameters. We explored two parameters – forest density and procedural algorithm – which both influence a large part of a forest's appearance. More specifically, we explored the impacts these two parameters have on the perceived believability and playability of generated forests. Believability and playability were chosen as they represent a desirable goal of procedural environment generation in games development, towards creating realistic immersive worlds, which are fun and engaging to play within.

Analysis of selection counts revealed that forests with a medium density were consistently chosen as the most playable and believable environments. This was also true across all image perspectives. It appears that if the aim of a game developer is to generate believable and playable forests, using a medium density produces the most optimal results. Another noteworthy result are the differences in selection between the first-person and aerial perspectives, with regards to forest density. In particular, there is considerably higher polarity between positive/negative votes from an aerial perspective. This indicates that participants could more easily determine the playability and believability of forests from an aerial perspective, as opposed to a first-person perspective.

Perhaps the most interesting result of the analysis of how the type of procedural algorithm affected selection counts, is that algorithms which were received positively in the first-person perspective were received negatively in the two aerial perspectives, and vice-versa. This is an unexpected result, as it signifies a considerable distinction and negative relationship between 1st-person and 3rd-person perspectives. This may be an interesting direction for further work in this area. In addition, both believability and playability selection counts displayed many similar patterns, with very little difference between the two cross-tabulations. This suggests that participants considered believability and playability very similarly, and perhaps implies a relationship between the two.

When comparing the cross-tabulations of forest density and procedural algorithm, there is also a clear distinction in terms of polarity. Specifically, the rankings of different forest densities contain far more polarised positive/negative votes than the type of generation algorithm. This shows that participants could more easily distinguish the playability and believability of forests with distinct densities, rather than distinct types of algorithm. These results may be of importance to the domain of procedural forest generation, since it highlights forest density has a more crucial role in creating forestry than previously expected. There is also a substantial contrast between 1st-person and aerial perspectives throughout our analysis, indicating that the perspective of the generated forest is an important consideration. This could inform future work and the games development sector of how to generate more realistic and engaging virtual forests. Furthermore, comparing how believability and playability are ranked shows considerable differences in polarity throughout.

8. Conclusions & Future Work

This paper presents a user study into virtual forests, using three different approaches of spatially distributing trees to approximate a plant community. These three approaches consisted of a random uniform distribution algorithm, a asymmetric plant competition model, and an iterative random distribution algorithm for creating clusters of trees. Through this study, the results demonstrate that the asymmetric plant competition model (the *'Propagation'* algorithm) produces forests which were rated the highest in terms of playability and believability, for both 2D and 3D aerial perspectives. This supports H1, suggesting that a bio-inspired plant competition model can produce forests which were rated the highest in these two criteria, but only for aerial image perspectives. This was not found in the case from a first-person perspective. Interestingly however, a method which geometrically approximates asymmetric plant competition using pseudo-randomness to distribute trees (the *'Clustering'* algorithm) received similar ratings for the same perspectives, and has utility as a less expensive alternative to plant competition models. We also found that the algorithms which score highly in the aerial perspective category were not scored as highly when viewed from the perspective of a player situated within the environment. Instead, we found that the control algorithm (pseudo-randomly distributing trees, the *Naive* approach) scored highly for both criteria when using this perspective. This may be advantageous to game designers who require an efficient alternative to expensive plant competition models. We also found a relationship between the forest density used in images and their rated playability by participants. In particular, forests generated with a high density scored low in playability but highly in realism – whereas forests generated with a low density scored low in realism and high in playability.

From this, we can say that if the objective of the environment designer is realism and playability, they must consider the perspectives in which the forest is to be viewed when deciding on a procedural algorithm to generate it. If for example, the virtual forest is to be used within a game where the player is situated within the forest, the *Naive* approach could be used to create satisfying content while simultaneously conserving computational resources. On the other hand, if the virtual forest to be created is to be used as scenery from an aerial perspective, then employing the asymmetric plant competition approach may generate more satisfying content.

Furthermore, the impacts of forest density and distribution algorithm on participant opinion were explored. More specifically, we were interested how these two parameters affected believability/playability selection frequencies. Several significant results were unearthed from analysing image selection counts, which may be of interest to game designers. For instance, forest density was found to be a more distinguishable characteristic than the type of procedural algorithm. In addition, forests generated with a medium density were consistently chosen as the most believable and playable distributions. These findings may inform both games developers and researchers of how to improve the quality of generated content. These findings support H2, that the canopy coverage (density) of generated forest images is a significant variable in how it perceived in terms of believability and playability.

In our experiments, our test group largely consisted of participants who were non-forest experts. One interesting area we would like to investigate in future work is the consideration of forest experts in our experiments. We could then contrast differences in preference between expert and non-expert viewpoints, which could offer some interesting insights. In addition, exploring the impacts of other visual characteristics of forestry is another aspect we are keen to develop in future work. For example, considering elements such as plant types, forest floor coverage, and other types of environment are all interesting questions we which to address through further investigation.

References

1. Condit, R.; Ashton, P.S.; Baker, P.; Bunyavejchewin, S.; Gunatilleke, S.; Gunatilleke, N.; Hubbell, S.P.; Foster, R.B.; Itoh, A.; LaFrankie, J.V.; others. Spatial patterns in the distribution of tropical tree species. *Science* **2000**, *288*, 1414–1418.
2. Williams, B.; Ritsos, P.; Headleand, C. Evaluating Models for Virtual Forestry Generation and Tree Placement in Games. Computer Graphics and Visual Computing (CGVC); Vidal, F.P.; Tam, G.K.L.; Roberts, J.C., Eds. The Eurographics Association, 2019. doi:10.2312/cgvc.20191259.
3. Togelius, J.; Kastbjerg, E.; Schedl, D.; Yannakakis, G.N. What is procedural content generation?: Mario on the borderline. Proceedings of the 2nd International Workshop on Procedural Content Generation in Games. ACM, 2011, p. 3.
4. Yannakakis, G.N.; Togelius, J. Experience-driven procedural content generation. *IEEE Transactions on Affective Computing* **2011**, *2*, 147–161.
5. Hastings, E.J.; Guha, R.K.; Stanley, K.O. Automatic content generation in the galactic arms race video game. *IEEE Transactions on Computational Intelligence and AI in Games* **2009**, *1*, 245–263.
6. Parish, Y.I.; Müller, P. Procedural modeling of cities. Proceedings of the 28th annual conference on Computer graphics and interactive techniques. ACM, 2001, pp. 301–308.
7. Togelius, J.; Champandard, A.J.; Lanzi, P.L.; Mateas, M.; Paiva, A.; Preuss, M.; Stanley, K.O. Procedural content generation: Goals, challenges and actionable steps. *Dagstuhl Follow-Ups* **2013**, *6*.
8. Doran, J.; Parberry, I. Controlled procedural terrain generation using software agents. *IEEE Transactions on Computational Intelligence and AI in Games* **2010**, *2*, 111–119.
9. Cordonnier, G.; Braun, J.; Cani, M.P.; Benes, B.; Galin, E.; Peytavie, A.; Guérin, E. Large Scale Terrain Generation from Tectonic Uplift and Fluvial Erosion. Computer Graphics Forum. Wiley Online Library, 2016, Vol. 35, pp. 165–175.
10. Shaker, N.; Togelius, J.; Nelson, M.J. Fractals, noise and agents with applications to landscapes. In *Procedural Content Generation in Games*; Springer, 2016; pp. 57–72.
11. Choroś, K.; Topolski, J. Parameterized and Dynamic Generation of an Infinite Virtual Terrain with Various Biomes using Extended Voronoi Diagram. *Journal of Universal Computer Science* **2016**, *22*, 836–855.
12. Hahn, E.; Bose, P.; Whitehead, A. Persistent realtime building interior generation. Proceedings of the 2006 ACM SIGGRAPH symposium on Videogames. ACM, 2006, pp. 179–186.
13. Martin, A.; Lim, A.; Colton, S.; Browne, C. Evolving 3d buildings for the prototype video game subversion. European Conference on the Applications of Evolutionary Computation. Springer, 2010, pp. 111–120.
14. Shaker, N.; Togelius, J.; Yannakakis, G.N.; Weber, B.; Shimizu, T.; Hashiyama, T.; Sorenson, N.; Pasquier, P.; Mawhorter, P.; Takahashi, G. The 2010 Mario AI championship: Level generation track. *IEEE Transactions on Computational Intelligence and AI in Games* **2011**, *3*, 332–347.
15. Williams, B.; Headleand, C.J. A Time-Line Approach for the Generation of Simulated Settlements. 2017 International Conference on Cyberworlds (CW). IEEE, 2017, pp. 134–141.
16. Bauer, S.; Berger, U.; Hildenbrandt, H.; Grimm, V. Cyclic dynamics in simulated plant populations. *Proceedings of the Royal Society of London B: Biological Sciences* **2002**, *269*, 2443–2450.
17. Lindenmayer, A. Adding continuous components to L-systems. In *L Systems*; Springer, 1974; pp. 53–68.
18. Prusinkiewicz, P. Graphical applications of L-systems. Proceedings of graphics interface, 1986, Vol. 86, pp. 247–253.
19. Pradal, C.; Boudon, F.; Nouguier, C.; Chopard, J.; Godin, C. PlantGL: a Python-based geometric library for 3D plant modelling at different scales. *Graphical models* **2009**, *71*, 1–21.
20. Shlyakhter, I.; Rozenoer, M.; Dorsey, J.; Teller, S. Reconstruction of plausible 3D tree models from instrumented photographs. *IEEE Computer Graphics and Applications* **2001**.
21. Tan, P.; Fang, T.; Xiao, J.; Zhao, P.; Quan, L. Single image tree modeling. *ACM Transactions on Graphics (TOG)* **2008**, *27*, 108.
22. Tan, P.; Zeng, G.; Wang, J.; Kang, S.B.; Quan, L. Image-based tree modeling. ACM Transactions on Graphics (TOG). ACM, 2007, Vol. 26, p. 87.
23. Livny, Y.; Yan, F.; Olson, M.; Chen, B.; Zhang, H.; El-Sana, J. Automatic reconstruction of tree skeletal structures from point clouds. *ACM Transactions on Graphics (TOG)* **2010**, *29*, 151.

24. Dale, H.; Runions, A.; Hobill, D.; Prusinkiewicz, P. Modelling biomechanics of bark patterning in grasstrees. *Annals of botany* **2014**, *114*, 629–641.
25. Desbenoit, B.; Vanderhaeghe, D.; Galin, E.; Grosjean, J. Interactive modeling of mushrooms. Eurographics (short papers). Eurographics, 2004, Eurographics Conferences, pp. 37–40.
26. Desbenoit, B.; Galin, E.; Akkouche, S. Simulating and modeling lichen growth. Computer Graphics Forum. Wiley Online Library, 2004, Vol. 23, pp. 341–350.
27. Reeves, W.T.; Blau, R. Approximate and Probabilistic Algorithms for Shading and Rendering Structured Particle Systems. *SIGGRAPH Comput. Graph.* **1985**, *19*, 313–322. doi:10.1145/325165.325250.
28. Gain, J.; Long, H.; Cordonnier, G.; Cani, M.P. EcoBrush: Interactive Control of Visually Consistent Large-Scale Ecosystems. Computer Graphics Forum. Wiley Online Library, 2017, Vol. 36, pp. 63–73.
29. Emilien, A.; Vimont, U.; Cani, M.P.; Poulin, P.; Benes, B. Worldbrush: Interactive example-based synthesis of procedural virtual worlds. *ACM Transactions on Graphics (TOG)* **2015**, *34*, 106.
30. Ecormier-Nocca, P.; Memari, P.; Gain, J.; Cani, M.P. Accurate Synthesis of Multi-Class Disk Distributions. Computer Graphics Forum. Wiley Online Library, 2019, Vol. 38, pp. 157–168.
31. Alsweis, M.; Deussen, O. Modeling and visualization of symmetric and asymmetric plant competition. Eurographics, 2005, pp. 83–88.
32. Hu, B.G.; De Reffye, P.; Zhao, X.; Yan, H.; Kang, M. Greenlab: A new methodology towards plant functional-structural model–structural part. Plant growth modelling and applications. TsingHua University Press and Springer, 2003, pp. 21–35.
33. Cournede, P.H.; Guyard, T.; Bayol, B.; Griffon, S.; De Coligny, F.; Borianne, P.; Jaeger, M.; De Reffye, P. A forest growth simulator based on functional-structural modelling of individual trees. Plant Growth Modeling, Simulation, Visualization and Applications (PMA), 2009 Third International Symposium on. IEEE, 2009, pp. 34–41.
34. Lane, B.; Prusinkiewicz, P. Generating spatial distributions for multilevel models of plant communities. Graphics Interface. Citeseer, 2002, pp. 69–80.