MASTER THESIS

Improving terrain generation using a single Generative Adversarial Network

Author:
Lars Sluijter

Supervisor:
Kevin Hutchinson

This thesis is submitted in fulfilment of the requirements for the degree of Master of Science in Game Technology

In the programme

Professional Master Game Technology
Academy for AI, Games and Media
Breda University of Applied Sciences

Date
June 25, 2023

Declaration of Authorship

I, Lars Sluijter, declare that this thesis titled, “Improving terrain generation using a single Generative Adversarial Network” and the work presented are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: 

Date: June 25, 2023

Breda University of Applied Sciences

Abstract

Academy for AI, Games and Media

Master of Game Technology

Improving terrain generation using a single Generative Adversarial Network

By Lars Sluijter

Previous research has shown that terrain generation using deep learning has promising results, but could be improved by combining heightmap generation and texture generation into one model. The aim of this research is to see what the effects of using a single GAN compared to two separate GANs are, in terms of realism, convergence, training time and computational performance. Computational performance includes both generation time and memory footprint as a metric. A new GAN was trained using datasets from previous studies to give an insight into these effects. This trained model is used to compare the results of this GAN to the GANs from these previous studies. Realism was evaluated by setting up a survey in which participants had to compare the output of the newly trained GAN with the output from previous GANs, and asking them to pick the one that looked most realistic to them. On top of that, the Fréchet Inception Distance (FID) was also measured for each of these GANs as a comparison. Convergence is compared by plotting the FID on intermediate models we acquire during training. Both the survey results and the FID scores indicated that the output of the new GAN produces more realistic results. The new GAN also converges faster than earlier proposed methods. The training time that the new GAN has been trained on is either worse or similar to earlier methods, although the new GAN does not have to train as long to get decent-looking results. The new GAN exhibits a much lower memory footprint than earlier methods, up to a 16 GB VRAM difference. The generation time is 2 milliseconds slower compared to earlier methods, although the generation time for these GANs were already low. The generation times for the new GANs, which were separately trained on two different datasets, are now approximately 18 milliseconds and 13 milliseconds.

Acknowledgements

I could not have done this research without the help of some people. I would like to dedicate this section to those people, and thank them for their help and contribution to this research.

I would first like to thank my supervisor, Kevin Hutchinson, for his help and support during my research. Weekly meetings with him helped me form my research, which was of great help, and this research would not stand as it is if it was not for his help. I would also like to thank my second reader, Flor Delombaerde, for giving feedback on my thesis occasionally. He also helped in giving me awareness into the statistical analysis part of my research, as well as giving tips to develop and work on my GAN. Furthermore, I would like to thank Luca Quartesan for giving some insight into deep learning methods, and giving me advice on some of the stuff I worked on during my research.

Lastly, I would like to thank my friends and family for the mental support they gave me during my research. They helped keep things fun and distract me from my work once in a while.

Contents

List of Figures

List of Abbreviations

DCGAN	Deep Convolutional Generative Adversarial Network
DEM	Digital Elevation Model
FID	Fréchet Inception Distance
GAN	Generative Adversarial Network
PCG	Procedural Content Generation
POI	Points-of-interest

Chapter 1: Introduction

Video games have become an increasingly important form of entertainment and a major industry in recent years. With advances in technology and game design, players now have access to a wide variety of games that offer engaging gameplay, immersive environments, and compelling stories. One of the key factors that contribute to the success of a game is its content, including levels, environments, items, and characters. However, creating such content can be a time-consuming and costly process, especially for games that require large amounts of content or have a high degree of variability. Procedural content generation (PCG) is a technique that has emerged as a promising solution to this challenge.

PCG is the creation of digital content using algorithms, and has been a developing field in the past decade. PCG can offer an increased variety in gameplay by facilitating the creation of game content using PCG algorithms. These algorithms can also ensure that video games can be replayable by generating new content each time the game is played. Video games such as *Minecraft* (Figure 1.1) (2011), *No Man's Sky* (2016), and dungeon crawl games such as *Diablo* (1997) and *The Binding of Isaac* (2011) have made use of PCG techniques, as each of these games have randomly generated levels.

Figure 1.1: Example of a Minecraft world, which is completely procedurally generated.

A recent addition to PCG is the use of deep learning. Deep learning provides new methods to generate a lot of different types of content used in video games, such as levels, textures, and music or sounds, as acknowledged by Liu et al. (2020). Liu et al. also acknowledges that recent developments in the field of PCG using deep learning have led to new opportunities and exciting advances in the field of PCG, such as generative adversarial networks (GANs), deep variational autoencoders and long short-term memory. Not only can deep learning be used for content generation, but also for content evaluation and gameplay outcome prediction, among other things.

One area in PCG that has been researched more recently with the use of deep learning is terrain generation. Studies done by Beckham and Pal (2017) and Panagiotou and Charou (2020) have made it clear that terrain can also be generated using deep learning, which can lead to more realistic looking terrain than making use of more traditional methods or algorithms for generating terrain. These studies make use of two separate generative adversarial networks (GANs) to generate a heightmap and a texture map for a terrain. However, these studies have also suggested that combining the scopes of the two GANs might make the terrain look more realistic. The drawbacks that may be raised by combining them are an increase of training time of the GAN, and a decrease in computational performance for generating a terrain.

This thesis dives into improving methods used by previous studies done in the field of terrain generation using deep learning, thus possibly increasing realism for the generated terrains, while also looking at the impact on the convergence, learning time, and computational performance of these deep learning models. The terrains that are generated by these deep learning models can then be used to yield a realistic 3D environment. These terrains can subsequently be used to help designers and artists to design a suitable terrain for their games, animations, and simulations. Increasing the realism of these terrains means that these designers and artists don't have to manually adjust the terrain a lot.

PCG can raise some ethical issues, as written by Cook, M. (2017). For example, PCG techniques can lead to undesired features, like unintentionally generating offensive pieces of content. The developer of such an algorithm needs to take this problem into account to prevent offensive content generation from happening. A similar issue is raised when scraping information from the internet to create a text generator. As the internet is huge, with lots of information, some offensive text might be generated when giving the generator some keywords. A ban list for words can prevent this issue. Another issue is that PCG as a label for a game is very popular with players, although this label can be abused as a selling point. For example, one might claim that making a million copies of a single weapon with different damage values is part of PCG, but this does not change anything about the gameplay itself.

Chapter 2: Background Information

Procedural content generation (PCG) is a method of creating game content automatically using algorithms, as opposed to creating content manually, as defined by Shaker, Togelius and Nelson (2016). This method of content creation extends to any type of content used in a video game, such as levels, quests, music and textures. An important component of PCG is that content generation takes the design and constraints of the game for which the content is generated into account. A key requirement of content generated by a PCG system is that it must be playable.

Because PCG can be used for a wide variety of content, PCG has lots of uses. According to Smith (2015), PCG can increase the replayability of a game by generating new levels or randomizing parts of the game. PCG can also be used to replace some of the human designers or artists in a game development company to create content using algorithms, following Shaker, Togelius and Nelson (2016). This can significantly reduce time and cost of the production of a video game, as well as support the designers or artists within the company by embedding these algorithms in intelligent design tools. Another use for PCG is to enhance the creativity of individual human creators. This means that small developers can create content-rich games without having to worry about creating all the content manually. To a further extent, as Shaker, Togelius and Nelson acknowledge, PCG can also be used to generate whole new games, possibly tailored to the player.

The research field surrounding PCG is growing fast, as the demand for PCG from game development companies is rising, according to the study by Hendrikx, Meijer, Van Der Velden and Iosup (2013). The study acknowledges that there has been significant progress in the field of PCG, although there is room for improvement in terms of detail, realism, and performance.

Although a research field on procedurally generated content has existed for a long time, procedurally generated content using deep learning is recent, according to Liu et al. (2020). Following the definition of the book by Kelleher (2019), deep learning is a subfield of artificial intelligence that focuses on creating multi-layer neural network models that can make accurate data-driven decisions. Deep learning is particularly suitable for problems where data is complex and large datasets are available. Because of these characteristics, deep learning has lots of uses. As an example, Kelleher's book states that deep learning models are used for speech recognition, face detection, image processing, and are also the core of self-driving cars.

Because of the characteristics declared in Kelleher's book, deep learning can also be used for PCG. Following the survey by Liu et al. (2020), previous research has shown that deep learning can provide new methods to generate levels, text, character models, textures, and music/sounds. As the types of content differ heavily, the generation of these types of content makes use of various deep learning methods depending on the content type.

One (sub)type of content that has been researched extensively within this research area is terrain generation. More classical ways to generate terrain use various forms of noise sources to help in generating terrain, as written by Beckham and Pal (2017) and Melnychuk (2020). These noise maps can be seen as heightmaps, which can then be used as a baseline for a terrain. The advantage of using noise is that it works quite fast to generate terrains, although it can also produce quite simple-looking terrains. Following Beckham and Pal's paper, harnessing the power of deep learning might lead to more interesting results.

A first step towards terrain generation using deep learning was done by Beckham and Pal (2017). In this study, two generative adversarial networks (GANs) were trained separately to achieve terrain generation in the form of a heightmap, which is generated by the first GAN, and a texture map, which is generated by the second GAN. According to Brownlee (2019), a GAN is a deep learning method where two deep learning models are trained, specifically a generator and a discriminator (Figure 2.1). Training works by making the generator generate a batch of samples based on random input vectors. This batch of "fake" samples is then mixed with real samples and are provided to the discriminator. The discriminator hereafter tries to classify whether each sample is real

or fake. Both models are updated afterwards, with the discriminator trying to discriminate better, and the generator trying to generate more real looking samples.

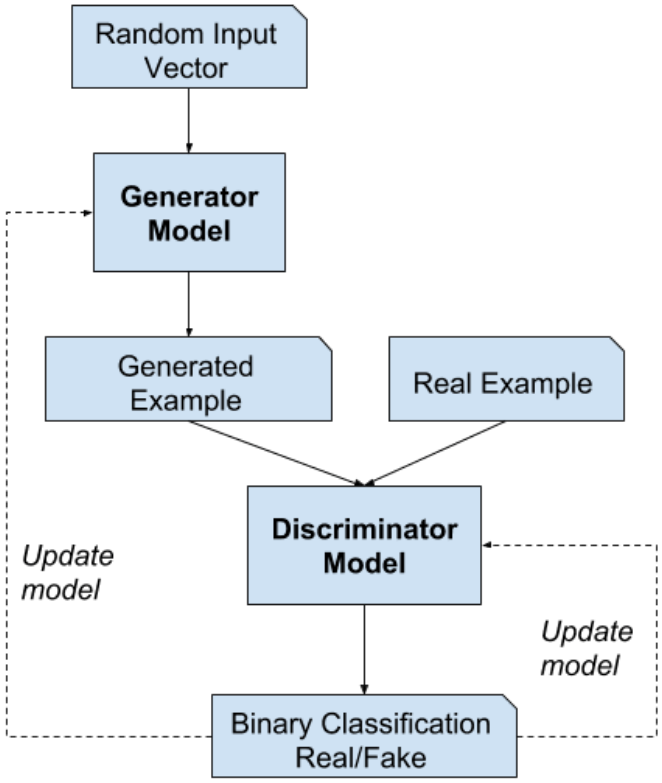


Figure 2.1: Example of the GAN architecture

The first GAN of Beckham and Pal's study (2017) generates a heightmap of the terrain. This heightmap is used as input by the second GAN afterwards, transforming the heightmap into a texture map (Figure 2.2). The GANs were trained on real-world data, resulting in realistic looking generated terrain. Because the two GANs were trained separately, some side-effects have occurred in the study, as parts of the generated texture were completely white. Beckham and Pal suggested that training the GANs jointly might resolve this problem and might possibly result in terrain that looks even more realistic.

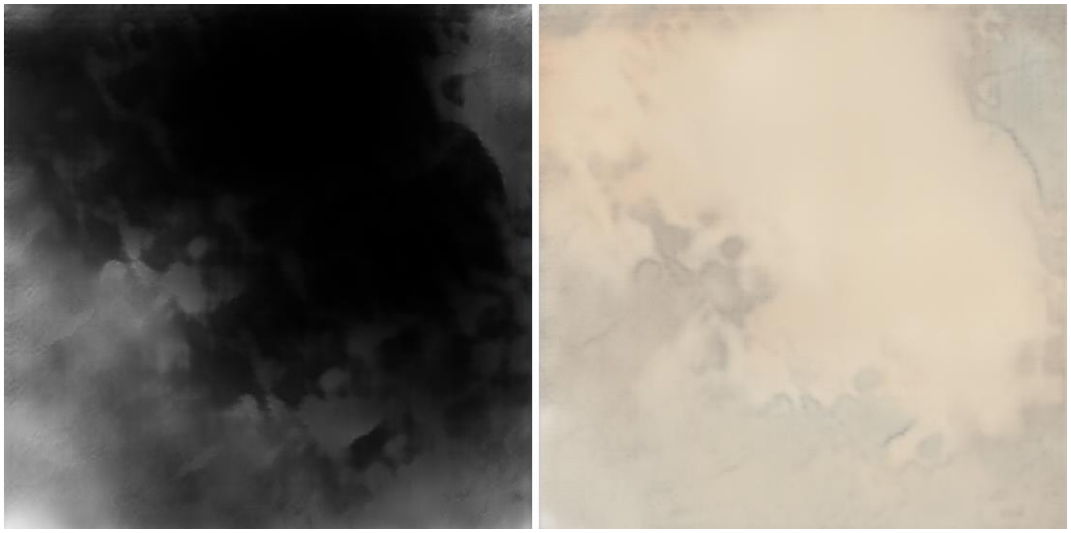


Figure 2.2: Terrain generation using Beckham and Pal's model, with on the left a randomly generated heightmap, and on the right the corresponding texture.

A similar study was done by Panagiotou and Charou (2020). This study covered the same approach as the previous study, except generation and transformation is done the opposite way; a satellite image is generated first, and a corresponding Digital Elevation Model (DEM) is generated afterwards (Figure 2.3). Both models used in the study were trained using satellite images from Greece, which means that the GANs produce terrain that looks like terrain around this area. While the results following this study are impressive, the authors claim, similar to the study done by Beckham and Pal, that more realistic and robust results might follow when combining the scope of both models. They do note that doing so might also lead to problems with convergence, meaning it could be difficult to train such a model.

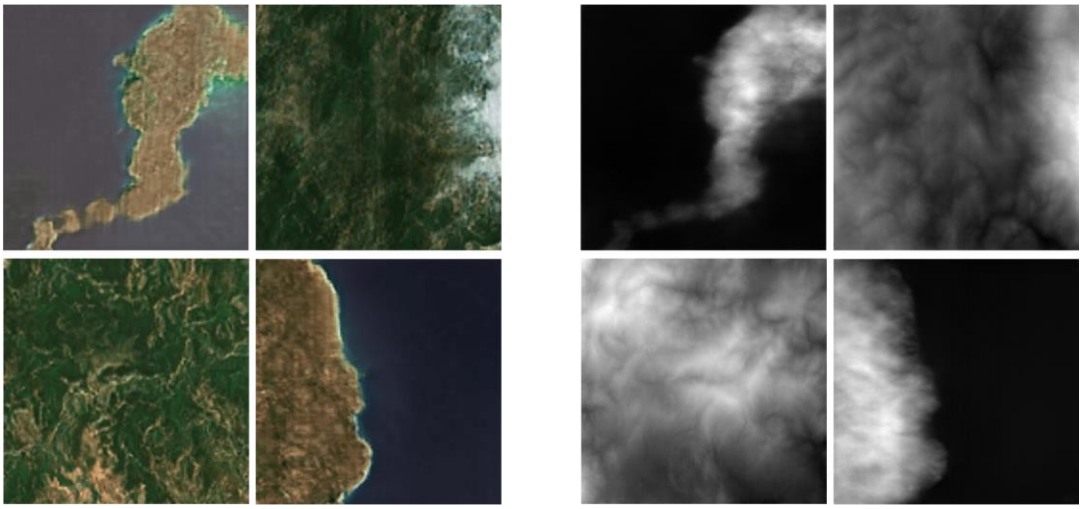


Figure 2.3: Satellite image generation using Panagiotou and Charou's model, with on the left randomly generated satellite images, and on the right the corresponding DEMs.

Other studies done in the field of terrain generation using deep learning only generate the heightmap of the terrain, like the studies done by Lopez-Garcia (2019) and Naik, Jain and Sharma (2022). Both studies use a single GAN to generate these heightmaps. A study done by Voulgaris, Mademlis and Pitas (2021) uses point-of-interest (PoI) scatter maps to generate realistic satellite terrain images containing geomorphological details (Figure 2.4). It is noteworthy that these scatter maps are not randomly generated, and need to be present beforehand to serve as input to the terrain image GAN.

Figure 2.4: Example of a PoI scatter plot (left) and its corresponding satellite image (right).

Following the study done by Beckham and Pal (2017), and the study done by Panagiotou and Charou (2020), it appears that combining the two models used in both studies might increase the realism of the generated terrain, while it might also lead to difficulty in the convergence of the deep learning model. Consequently, a deep learning model of higher complexity is necessary, which will take a longer time to train, and will probably also lead to a decrease in computational performance, meaning that the generation of terrain will take longer, and a larger memory footprint. Consequently, this thesis aims to give an answer to the following research question: what are the effects of using a single GAN to generate both a heightmap and texture for terrain compared to using two separate GANs, in terms of realism, convergence, training time, and computational performance?

Chapter 3: Methodology

3.1 Metrics

To fully give an answer to the research question declared in the previous section, some metrics must be measured and evaluated to reach a conclusion. These metrics are:

- Realism.
- Convergence.
- Training time.
- Computational performance.

The chosen metrics flow from the previous studies by Beckham and Pal (2017) and Panagiotou and Charou (2020). Both studies claim that realism should increase when using a single GAN. Panagiotou and Charou also mention that combining the scopes of both GANs used in their study could lead to difficulty in convergence. Therefore, it is also important to look at training time and computational performance as the complexity of the GAN increases. This is because the resulting GAN might be used in the field. Note that computational performance includes both generation time and memory footprint as a metric.

3.2 Developing the GAN

Before being able to evaluate the metrics mentioned in the previous section, a single GAN must be developed for the purpose of terrain generation. As the method of generating terrain uses only a single GAN, as opposed to earlier studies which used two GANs, the GAN will be named SingleTerrainGAN from this point forward in this thesis.

The architecture of the GAN is taken from a study done by Karras et al. (2020). In this paper, the *StyleGAN2* architecture is introduced, which is an improvement to the StyleGAN architecture earlier introduced by Karras et al. (2019). During the training process of StyleGAN2, a set of "style" vectors manage the key aspects of the image that is generated, such as its colour, texture, and shape. These style vectors are adjusted during the training process to generate realistic images that resemble the training data. Instead of epochs, which are usually used as the training iteration metric for GANs, StyleGAN2 uses k-imgs. K-imgs is the number of training images that the network has been trained on, measured in thousands.

The difference between the architecture used in the study by Karras et al. and the architecture used for SingleTerrainGAN, is that SingleTerrainGAN uses 4 channels for its images instead of the usual 3 channels. This includes the red, green, and blue channels for texture colours, and has an extra channel for the heightmap. An implementation by Diego Porres[1] is used to achieve this. This implementation makes use of the PyTorch library, which is a well-known Python library for deep learning practices.

As the research question requires the comparison of the developed single GAN with the other GANs used in previous studies, these GANs also need to be replicated to be able to get a full comparison.

The GANs that were developed by Beckham and Pal (2017) are open source. The study uses a Deep Convolutional GAN (DCGAN) to generate heightmaps, and a pix2pix architecture to translate this heightmap to a texture. The GANs use the Lasagne python library, combined with Theano. As both libraries are discontinued, a decision was made to recreate the architecture of these GANs fully in TensorFlow, another deep learning library that is popular in the field of deep learning. As TensorFlow tries to allocate as much GPU memory as possible during training or generation, memory growth is turned on. This reduces the memory that is allocated by only allocating as much memory as is necessary.

The GANs that were developed by Panagiotou and Charou (2020) are open source as well. The study uses a ProGAN to generate the texture of the terrain, and a pix2pix architecture to translate this texture to a heightmap. The ProGAN was developed using the PyTorch python library, while the pix2pix GAN was developed using TensorFlow. Because PyTorch is a widely supported library at the moment of writing this thesis, these GANs will be trained and used as-is, as opposed to the GAN that was developed by Beckham and Pal.

3.3 Datasets

To be able to train SingleTerrainGAN, one or multiple datasets are necessary. The datasets that are used to train SingleTerrainGAN are derived from the datasets used by the studies done by Beckham and Pal (2017) and Panagiotou and Charou (2020). These studies used satellite images from deserts and from Greece, respectively.

[1] https://github.com/PDillis/stylegan3-fun

As the two datasets heavily differ from each other in terms of content, SingleTerrainGAN is trained on these two datasets separately, thus preventing the generator from generating terrain with mixed biomes. This is also better for comparing the GANs, as we can directly check the difference in output from two GANs using the same dataset. This is also necessary because the images in these two datasets differ in size; the desert dataset has images with a size of 512x512 pixels, while the Greece dataset has images with a size of 256x256 pixels.

The desert dataset will be augmented using the ImageDataGenerator class available for TensorFlow beforehand, which leads to larger datasets that can be used to train the GAN. This is done because the desert dataset only contains 240 images, which is not a lot to train on. The Greece dataset on the other hand contains more than 1665 images, which should be more than enough to train on. The desert dataset will be filled with augmented data until reaching a dataset with 2000 images.

A full overview of the GANs that are compared in this research can be seen in table 3.1.

GAN	**Architecture**	**Library used**
Beckham and Pal (2017), recreated	DCGAN (heightmap) + pix2pix (texture)	TensorFlow
Panagiotou and Charou (2020)	ProGAN (texture) + pix2pix (heightmap)	PyTorch (ProGAN) TensorFlow (pix2pix)
SingleTerrainGAN	StyleGAN2	PyTorch

Table 3.1: Overview of GANs that are compared in this research.

3.4 Training the GANs

The GANs are trained on a dedicated rented GPU that includes an NVIDIA GeForce RTX 3070.

SingleTerrainGAN will be trained for 5000 k-imgs. With this amount of k-imgs, we ensure that SingleTerrainGAN has enough time to sufficiently train on the datasets. To counteract any possible noise that may be present in the generated heightmaps, a box blur with a radius of 1 will be applied to the heightmaps generated by SingleTerrainGAN. This means that for each pixel, the average value of the pixel itself and the neighboring pixels will be taken. This smoothens the heightmap a bit. An

example of the use of box blur in noisy terrains can be seen in figure 3.1. The other GANs will be trained following the settings set by their authors.

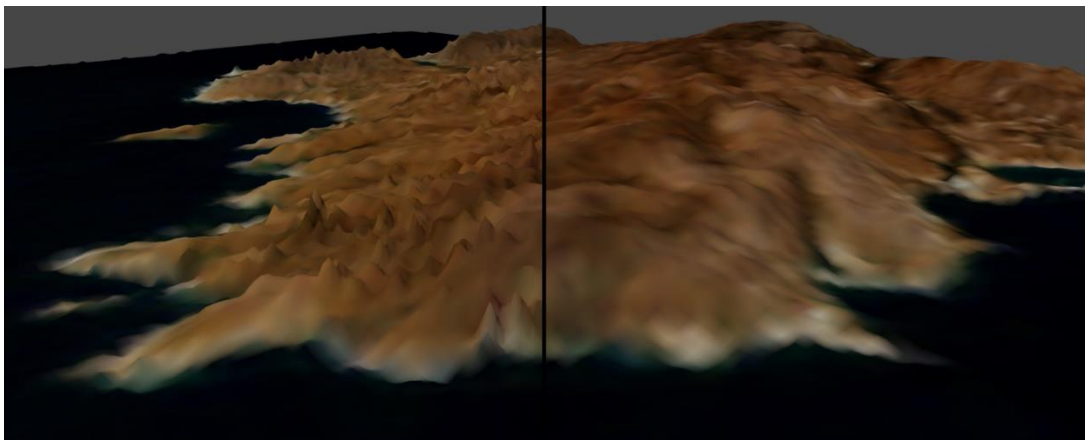


Figure 3.1: Left: terrain without blur, right: terrain with box blur.

3.5 Evaluating realism for the generated terrain

To evaluate whether terrains that are generated by SingleTerrainGAN are more realistic than terrains generated by GANs from previous studies, SingleTerrainGAN must be compared to the GANs that were used in the studies by Beckham and Pal (2017) and Panagiotou and Charou (2020). There are a multitude of ways to evaluate GANs in the form of quantitative metrics and qualitative methods.

3.5.1 Quantitative metric

Quantitative metrics can be applied to assess the degree of resemblance between the generated samples of a GAN and the source images it was trained on. A list of quantitative metrics employed for this purpose are listed in papers by Borji (2018) and Borji (2022). Although the degree of resemblance does not fully encapsulate the definition of realism, it is good to look at how the outcomes of these quantitative metrics differ between the different GANs.

The quantitative metric that will be used for evaluation is *Fréchet Inception Distance (FID)*. This metric is one of the most used metrics for evaluating GANs, and is considered to be more established than other metrics, following a paper by Slangewal (2019). This metric is also proven to work well in terms of discriminability, robustness and efficiency, following a paper by Xu et al. (2018). The metric uses a classifier, and compares the values of neurons of an intermittent layer. These values are combined into a distribution, one for generated samples, and one for real samples. The distance

between these two distributions is then calculated using the Fréchet Distance technique. The smaller this distance is, the higher the GAN scores using this metric. The Python-based *pytorch-fid* library is utilized to compute the FID of the GANs, by comparing 10.000 generated terrains against the original dataset.

By getting the FID score for each GAN, we can compare the values to see whether one GAN scores better than others, which may indicate a difference in realism of the generated terrain.

3.5.2 Survey

Because the quantitative metrics chosen to evaluate SingleTerrainGAN are not enough to get a full view of whether SingleTerrainGAN generates more realistic looking terrain, a more manual manner of evaluation is also adopted to assess this.

The method used is an adapted version of the *Ratings and Preference judgment* method, mentioned in the paper by Borji (2018). By using this method, a series of side-by-side (generated) images are shown to participants in a survey. The participants of the survey have to compare the images, and choose which image they prefer in terms of realism. Although this method has been used before to evaluate GANs in earlier studies, the method has not been used for generated terrain yet.

The intended participants for the survey are the general population. This is because we need to know whether a general audience could see if the terrains produced by SingleTerrainGAN seem more realistic compared to the other GANs. The survey is also directed towards people with knowledge or experience in terrain design, as these people may give slightly different answers than a general audience, which might be interesting to analyse.

The survey starts off by giving the participants information on the content of the survey, and its purpose. The participants are also informed that submission is completely anonymous. The participants are then asked how familiar they are with the look and feel of the different types of terrain that are shown in the survey, and how frequently they have visited a place with this type of terrain. The participants must answer these questions on a scale of 1 to 5. As we want to evaluate whether the generated terrains seem more realistic to a general audience, we want to see whether people that are more familiar with a certain type of terrain differ from opinion against the general audience.

The participants are then asked whether they have any experience or background in the field of terrain design. If this is the case, they will be asked on a 1 to 5 scale how familiar they are with terrain design of deserts, and are asked the same for Mediterranean landscapes. This is done to see whether there is a correlation between the knowledge of a participant in terrain design, and their answers chosen during the survey.

After filling in this background information, the participants will be given 20 side-by-side comparisons between terrains generated by SingleTerrainGAN, and terrains generated by other GANs and real terrain samples taken from the training datasets. The participants are asked to pick one of the terrains that appear most realistic to them. These comparison questions are shown in a random order, and are picked randomly from a list of a total of 30 questions. These 30 questions are populated with 10 different comparisons between terrains generated by each GAN from previous studies, and terrains generated by SingleTerrainGAN. The other 10 comparison questions contain a comparison between satellite images and terrains generated by SingleTerrainGAN. The participants are also asked how confident they are in their answer on a scale of 1 to 5. This gives an insight into whether there is a big difference in perceived realism between the two terrains, or whether people are less confident in their answers and the perceived realism is similar. The confidence rate is measured separately as this makes statistical analysis easier.

Participants are also asked what properties of the terrain they looked at when choosing which terrain is more realistic. The properties the participants can choose from are colour, height, whether there are any artefacts present, and whether there are patterns visible in the terrain. The participants are also able to fill in their own properties they looked at when making their decision. This gives an insight into what participants look at when making their decisions.

This will provide sufficient information to draw a conclusion regarding the realism of terrains generated by different GANs, while also accounting for the time that survey participants are required to invest in filling out the survey. Because letting an algorithm choose random terrains for these questions could lead to questions that would seem incomparable, for example landscapes with totally different features, a manual approach is chosen. The terrains that are used for the comparisons are hand-picked, and are selected based on how similar the terrains from the different sources look.

The terrains that will be used and compared in this survey need to be visualized in some way. This is done by using a 3D rendering engine. This study uses Blender to model the terrain using a texture and a heightmap. To achieve this, the terrain texture is put on a plane object, and the heightmap is used on a displace modifier. Because of the different sizes that are output by the GANs, the strength of this displace modifier differs per terrain size. The strength is set to 0.05 for terrains with a size of 512x512, and 0.2 for terrains with a size of 256x256.

Subsequently, the model is exported to a glTF format, and rendered in *3dviewer.net* to visualize the terrains for the survey participants. 3dviewer.net is an online 3D model viewer that allows the user to zoom, rotate and pan on a 3D model. The 3D model viewer can also be embedded in the survey, meaning the survey participants can get a detailed look at the terrains when comparing them without having to open any links or leave the survey. An example of a terrain rendered using 3dviewer.net can be seen in figure 3.2.

Figure 3.2: Example render of a terrain in 3dviewer.net.

3.6 Evaluating GAN performance

The research question includes metrics that measure and compare the performance of the GANs. Convergence, training time and computational performance are all quantifiable metrics which can be measured during training of the GANs, or during the generation of the terrains.

To compare the convergence of the different GAN architectures, the intermediate FIDs of the model are used instead of looking at the loss during the training process. This is because of the way a GAN works: because we have a generator and discriminator that are trained against each other, the loss of both will go up or down irregularly, meaning that this is not a good way to see how the model converges. Diagrams are made to visualize the intermediate FID values of the generator throughout the training process. This is visualized by a line diagram, with on the X-axis the number of epochs or k-imgs, and the FID value on the Y-axis.

After visualizing the convergence using line diagrams, these diagrams will be compared through observation. By determining at what epoch or k-imgs there is no big increase or decrease in performance, and by comparing the lowest point in the graph, conclusions can be drawn on the convergence of each of these GAN architectures.

The training time is tracked while training the models. For the GANs used for the study done by Panagiotou and Charou (2020), the GANs are trained separately from each other, which means that the training time will be the sum of the training time of both individual GANs.

Measuring the generation time for each of the GANs is done by generating 50.000 terrains for each GAN, and tracking how much time it takes to generate. Afterwards, the mean average of generating one terrain will be taken as the generation time. Terrains will be generated in batches of 16.

Measuring the memory footprint is done by using the NVIDIA System Management Interface (nvidia-smi) to check the GPU memory usage while generating terrain textures/heightmaps using the trained models. This can give a representation of the peak GPU memory usage for each model when generating. These values can be compared afterwards to see how SingleTerrainGAN compares to the other models in terms of peak GPU memory usage. An example of the use of nvidia-smi can be seen in figure 3.3. In this figure, the value that will be recorded for each model is highlighted.

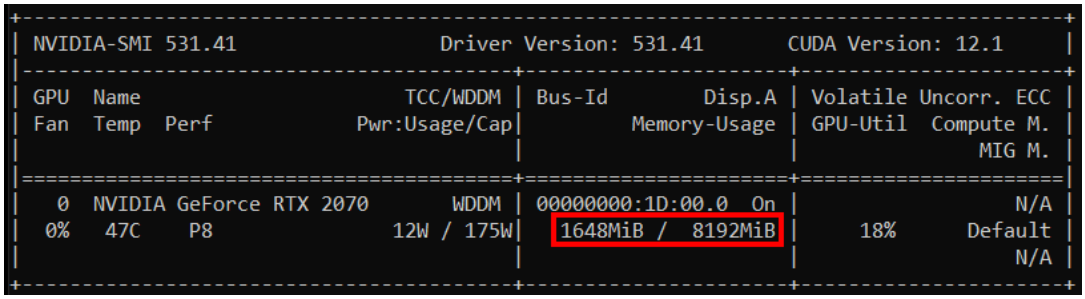


Figure 3.3: Example of nvidia-smi, with GPU memory usage highlighted in a red rectangle.

Chapter 4: Data and Results

4.1 Trained GAN output

Before getting into detail on the data and results that were gathered from the survey and the quantitative metrics, we will have a look at some example terrains that are generated by SingleTerrainGAN. In figure 4.1 you can find a series of rendered terrains generated by SingleTerrainGAN.

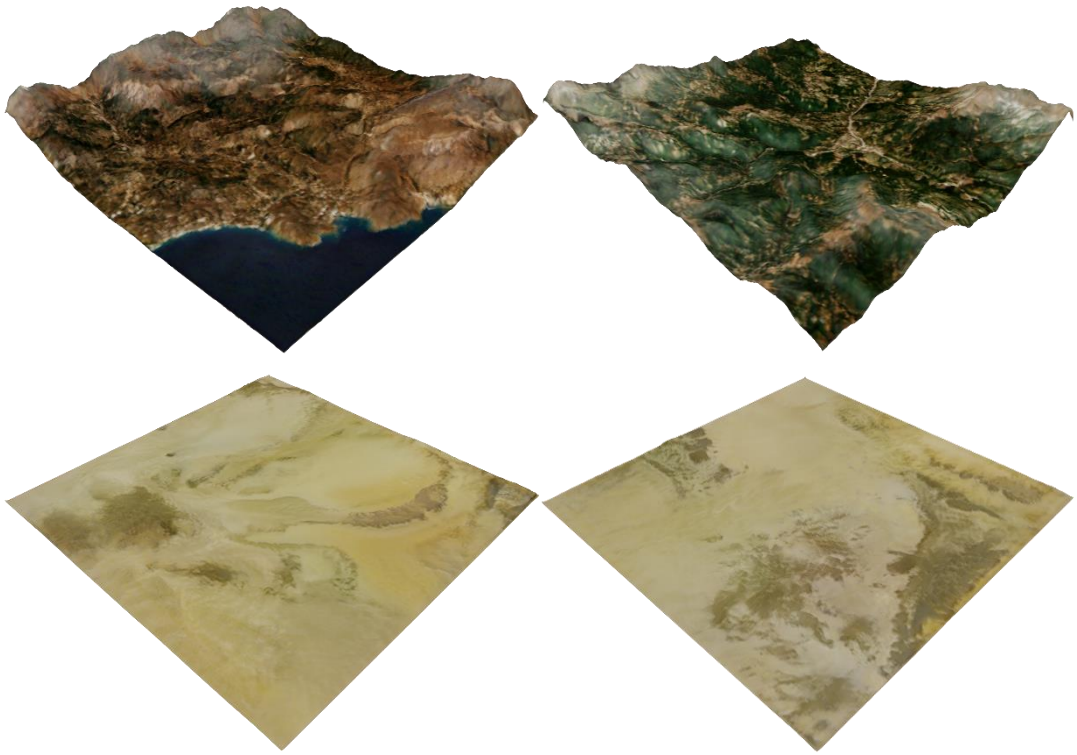


Figure 4.1: Series of rendered terrains generated by SingleTerrainGAN.

4.2 Survey

The purpose of the survey is to determine if there is a difference in perceived realism of terrains generated by the old models, satellite images, and terrains generated by SingleTerrainGAN. The survey consists of a series of terrain comparisons, as stated in chapter 3. The survey also contains questions about some background information of the participants; that is the familiarity and experience with the different types of terrains that are compared, and whether the participant has any expertise in terrain design.

The survey received 51 responses which were fully filled in, and 23 responses that were partially filled in. Participation in the survey was completely voluntarily without any reward, and could be filled in online through a browser. The partially filled in responses are still used in the evaluation of the survey, as the data that was filled in is still usable.

An overview of all terrains that were compared in the survey can be found in appendix A. The terrains on the left in these comparisons are terrains that were generated by one of the previous models, or satellite images. The terrains on the right in the comparisons are terrains that were generated by SingleTerrainGAN. Note that the order in which these terrains were shown in the survey was randomized, such that the participants had no way of knowing the source of the terrains. The prefix names of these comparison questions are based on the dataset they are generated from, “D” meaning desert, and “G” meaning Greece. On top of that, an additional “S” indicates that a comparison to a satellite image is made.

4.2.1 Comparison questions

Each survey participant was randomly assigned twenty comparison questions, regardless of the type of terrains that were compared. During these questions, the participants were asked to pick the terrain that appeared most realistic to them. Besides this, the participants were also asked to rate their level of confidence in their choice on a scale from 1 to 5.

The number of times a certain terrain has been chosen over another in a comparison question can be seen in figure 4.2, figure 4.3 and figure 4.4. Because it is also interesting to look at the mean confidence of the participant for each comparison question, these are given in figure 4.5, figure 4.6 and figure 4.7. These figures also include the mean confidence for each different choice. The figures are split up between the different comparison sets.

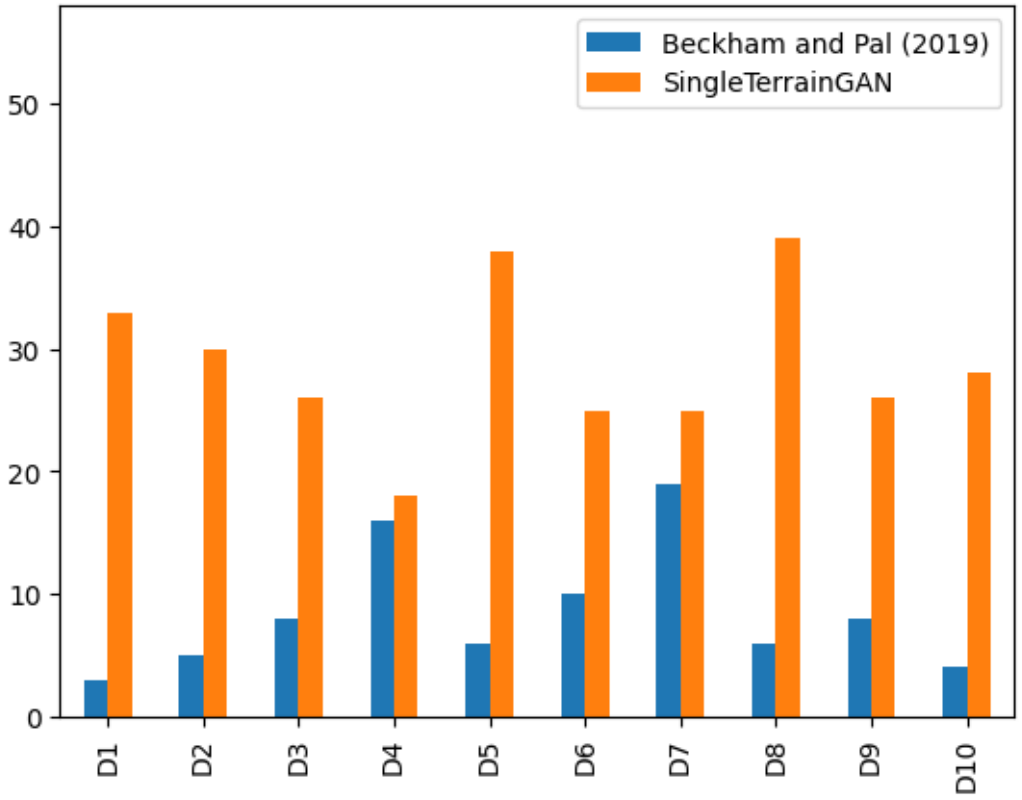


Figure 4.2: Frequency of picked options for 10 different comparisons between terrains generated by Beckham and Pal's (2017) model and SingleTerrainGAN.

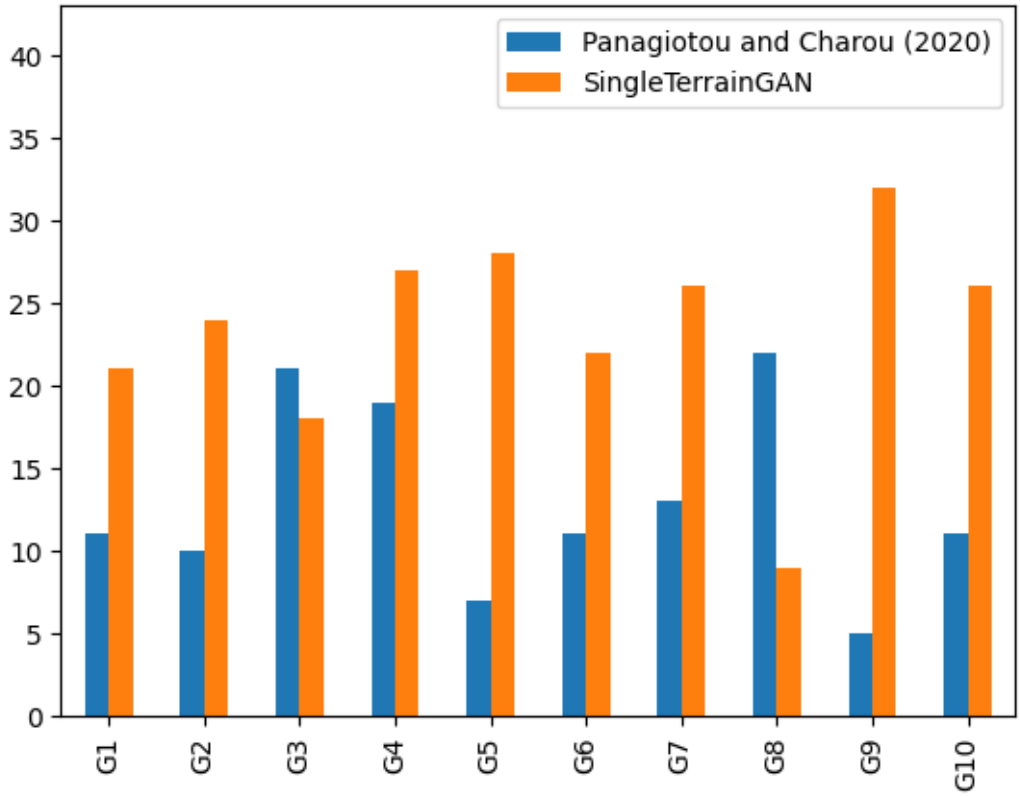


Figure 4.3: Frequency of picked options for 10 different comparisons between terrains generated by Panagiotou and Charou's (2020) model and SingleTerrainGAN.

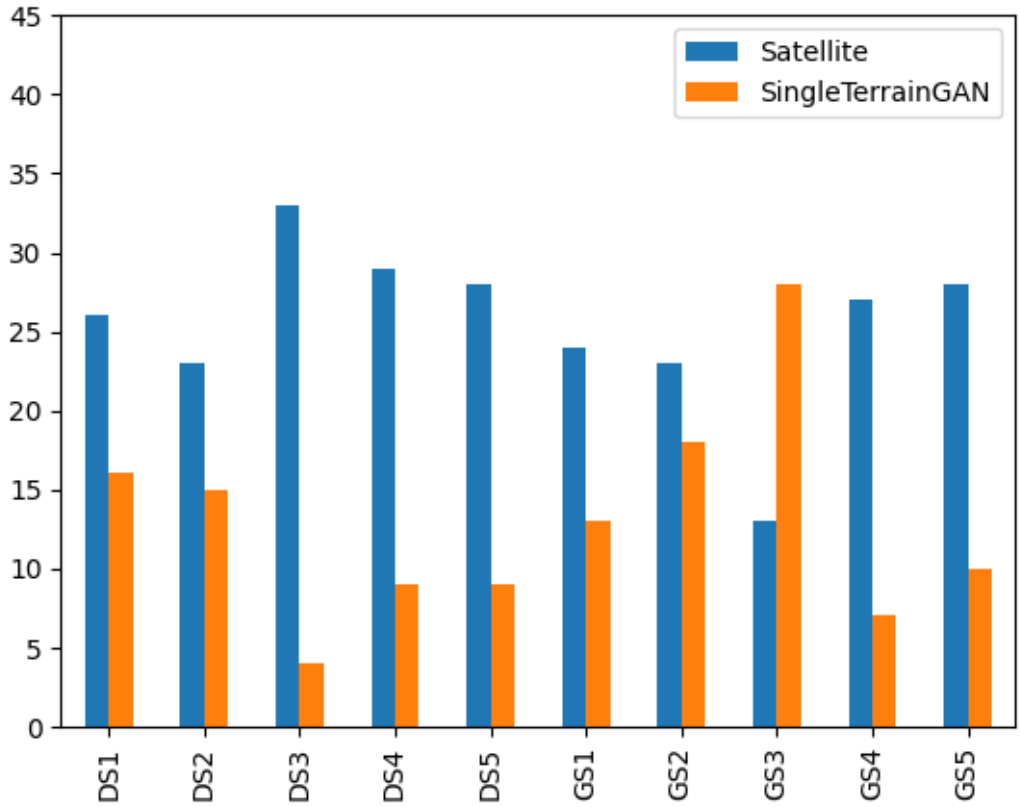


Figure 4.4: Frequency of picked options for 10 different comparisons between terrains rendered from satellite images and terrains generated by SingleTerrainGAN.

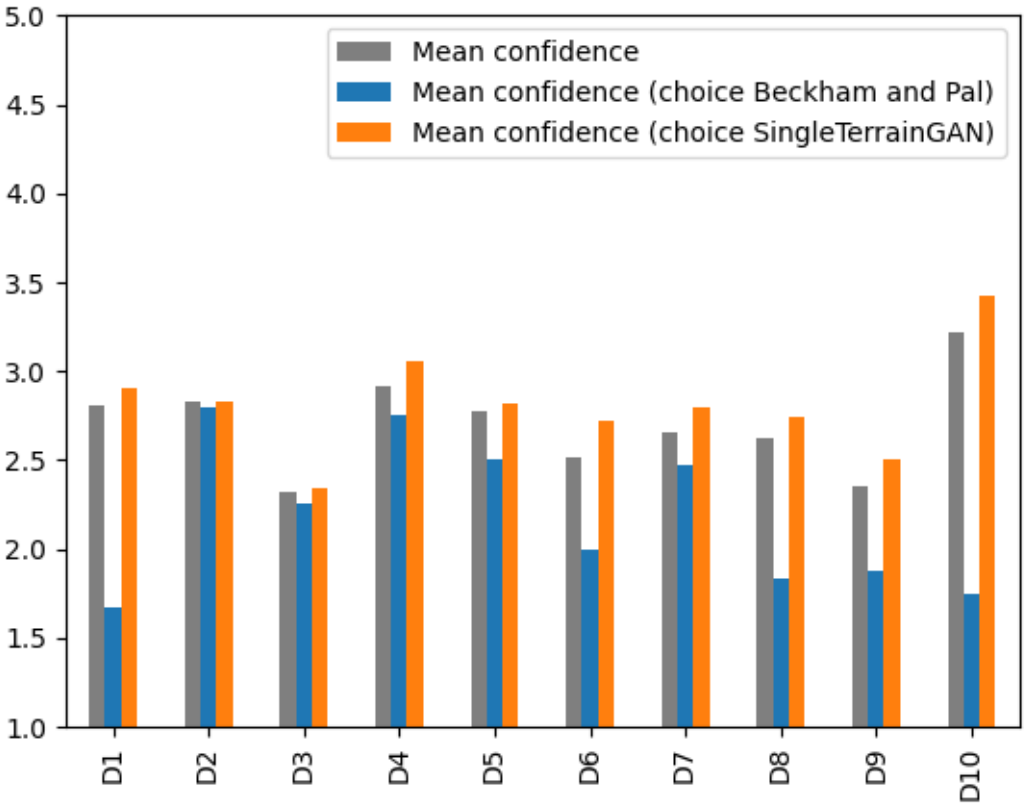


Figure 4.5: Mean confidence rate of every desert comparison question, including the mean confidence of each individual choice.

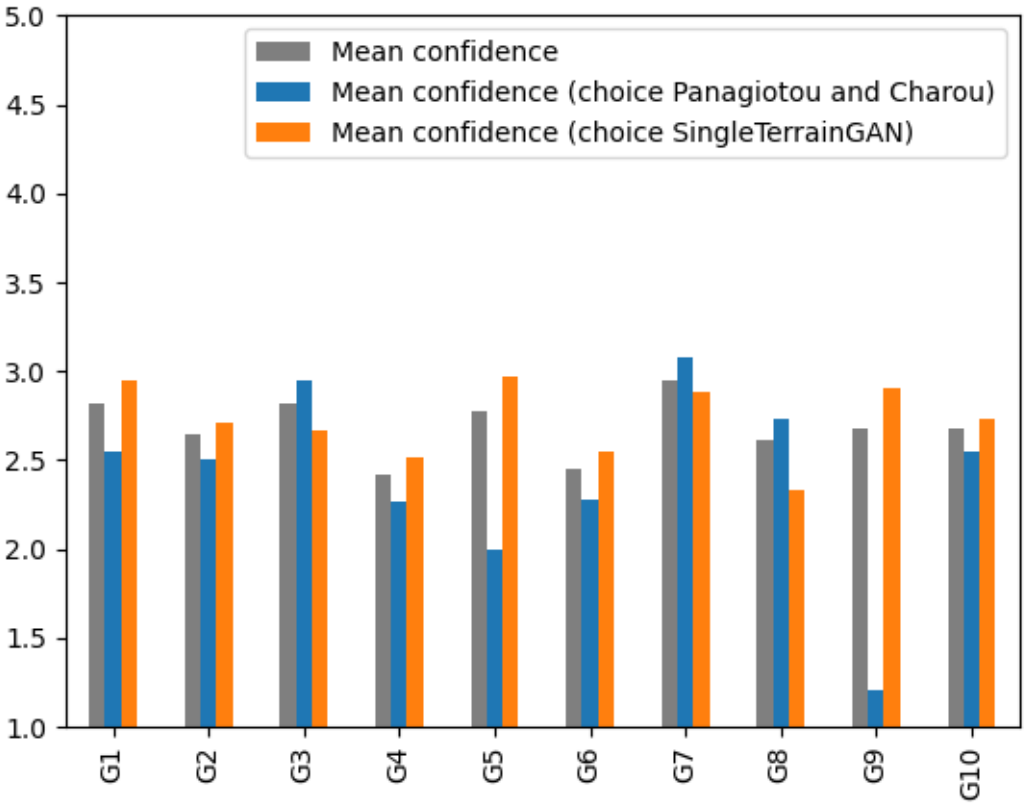


Figure 4.6: Mean confidence rate of every Greece comparison question, including the mean confidence of each individual choice.

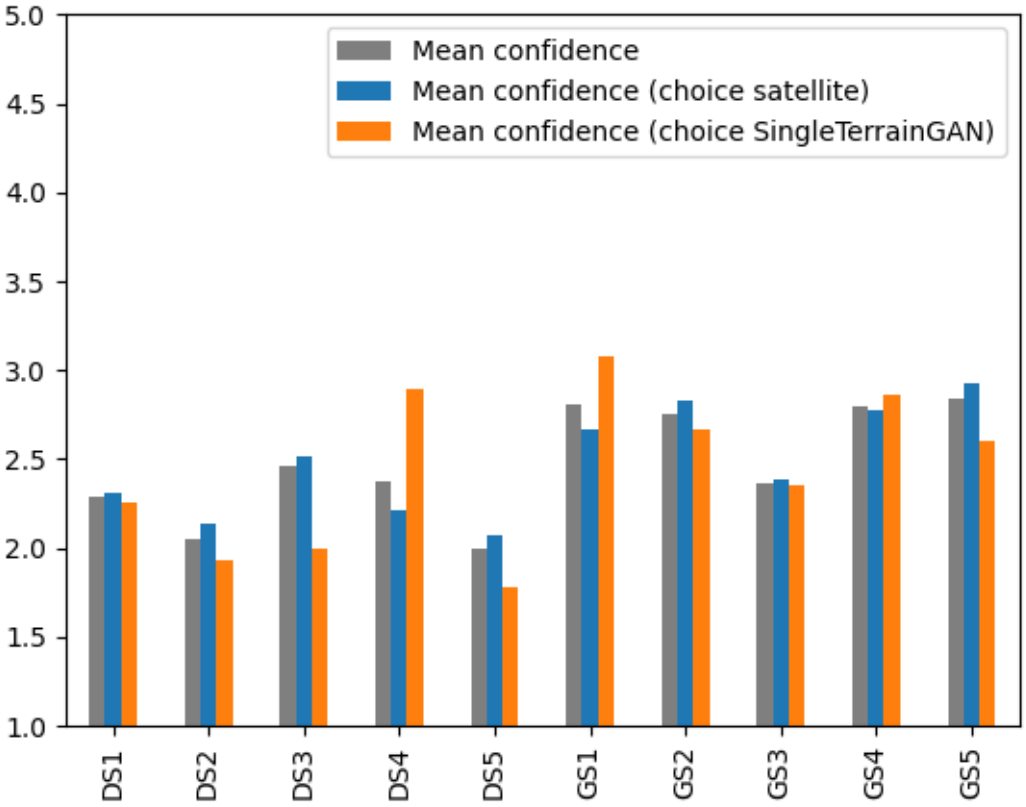


Figure 4.7: Mean confidence rate of every satellite comparison question, including the mean confidence of each individual choice.

In these graphs, it is apparent that for the desert comparison set, terrains generated by SingleTerrainGAN have been chosen more often than the terrains generated by Beckham and Pal's (2017) model. The mean pick rate of every desert terrain comparison, with 0 corresponding to Beckham and Pal's model and 1 corresponding to SingleTerrainGAN, is 0.774. The mean confidence rate for all desert terrain comparison questions is 2.692. It is also noteworthy that the mean confidence rate was consistently higher for the SingleTerrainGAN choice.

For the terrain comparisons based on the Greece dataset, the results are a little closer. Terrains generated by Panagiotou and Charou's (2020) models have been chosen over terrains generated by SingleTerrainGAN in a few comparisons, and some comparisons are close in terms of how frequently the terrains were picked. The mean pick rate of every Greek terrain comparison is 0.637. The mean confidence rate for all Greek terrain comparison questions is 2.694. The mean confidence rate for the SingleTerrainGAN choice in these comparisons was always higher in the questions where the SingleTerrainGAN choice was chosen more often, except for question G7. In this case, the

other option had a higher mean confidence rate, although this option was chosen less often than the SingleTerrainGAN option.

For the terrains that were compared against satellite images, it is apparent that in most cases the terrains rendered from satellite images were chosen more often, except for one comparison. The mean pick rate for these comparisons is 0.327. The mean confidence rate for these comparison questions is 2.474.

To find out whether the difference in choices made by the participants has any statistical significance, a chi-square test is performed on the frequency count of the choice data of each comparison set, categorised by choice. For this test, a null hypothesis is set up. This null hypothesis posits that there is no significant difference in perceived realism between the terrains generated by the two different models. An alternative hypothesis is also set up, which says that there is a significant difference in perceived realism between the terrains generated by the two different models. The chi-square test outputs a p-value that can be used to determine whether the null hypothesis can be rejected. The p-values output by the chi-square tests for each comparison set is lower than 0.01.

To see whether the confidence of participants changes during the duration of the survey, the mean confidence over the course of the survey is plotted in figure 4.8.

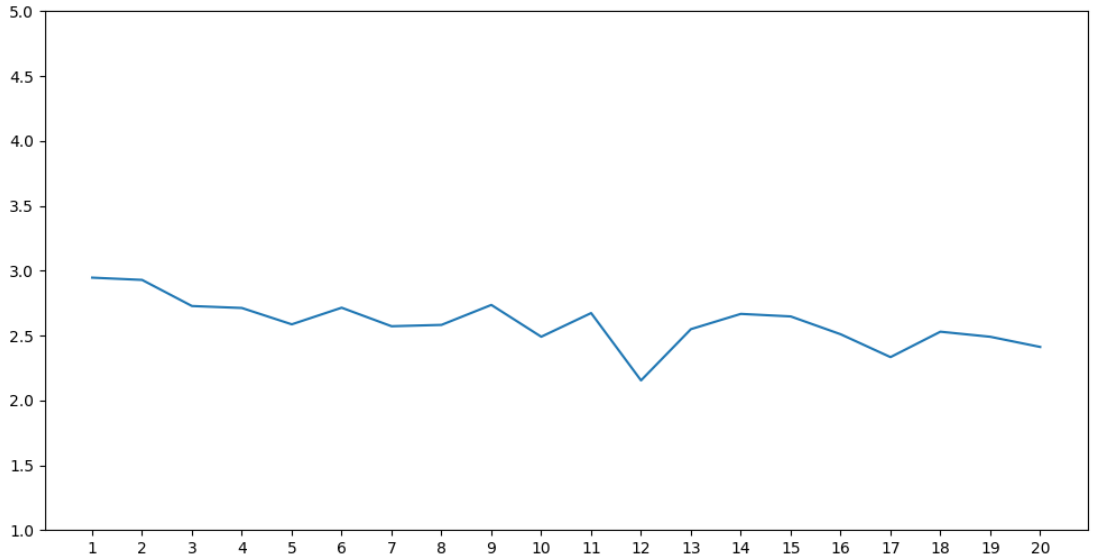


Figure 4.8: Mean confidence plotted over the duration of the survey.

In figure 4.8, the mean confidence level of participants is at its highest point at the first filled in question. The mean confidence then drops down a little in later questions, and drops and rises irregularly at the second half of the survey. On top of this statistic, the number of times a confidence level of 1 has been chosen is displayed in figure 4.9.

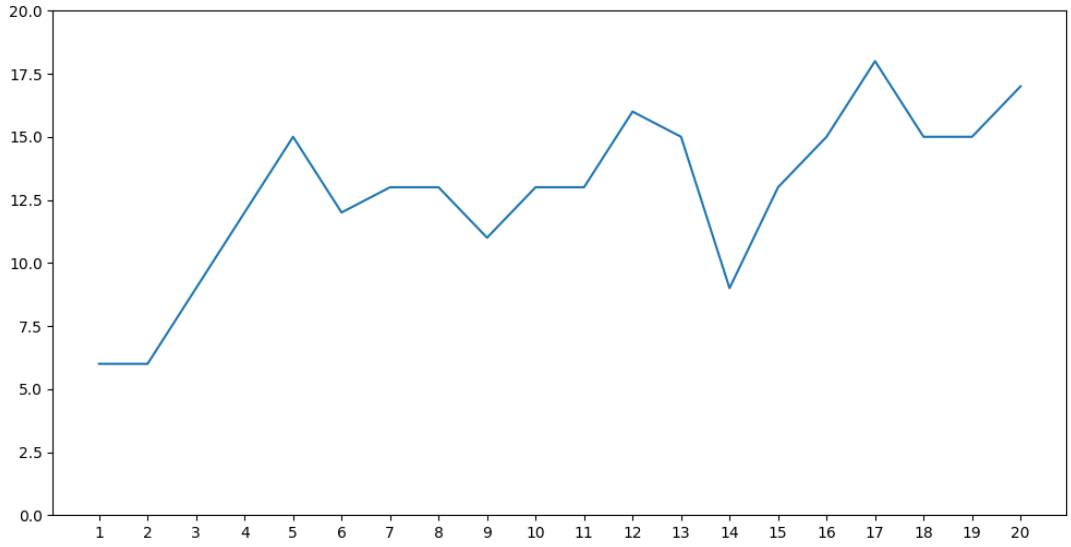


Figure 4.9: Frequency count of the number of times a confidence level of 1 has been chosen.

4.2.2 Reasoning

For each comparison question, participants also had to indicate what their reasoning was for picking one terrain over another. This was done by having them choose from a fixed set of options, as well as the possibility to choose an "other" option. This option allowed participants to indicate what their reasoning was themselves if their reasoning did not fit one of the fixed options.

A frequency chart of the options chosen per comparison question can be found in figure 4.10 for the desert comparisons, figure 4.11 for the Greece comparisons, and figure 4.12 for the comparison questions between satellite images and generated terrains. Furthermore, the mean average of times that each reasoning has been chosen is highlighted in figure 4.13, and a choice-grouped bar graph with this same mean average can be found in figure 4.14. Choice 1 in this case indicates either terrains generated by the GANs developed by Beckham and Pal (2017) or Panagiotou and Charou (2020), or a satellite image. Choice 2 indicates a terrain generated by SingleTerrainGAN.

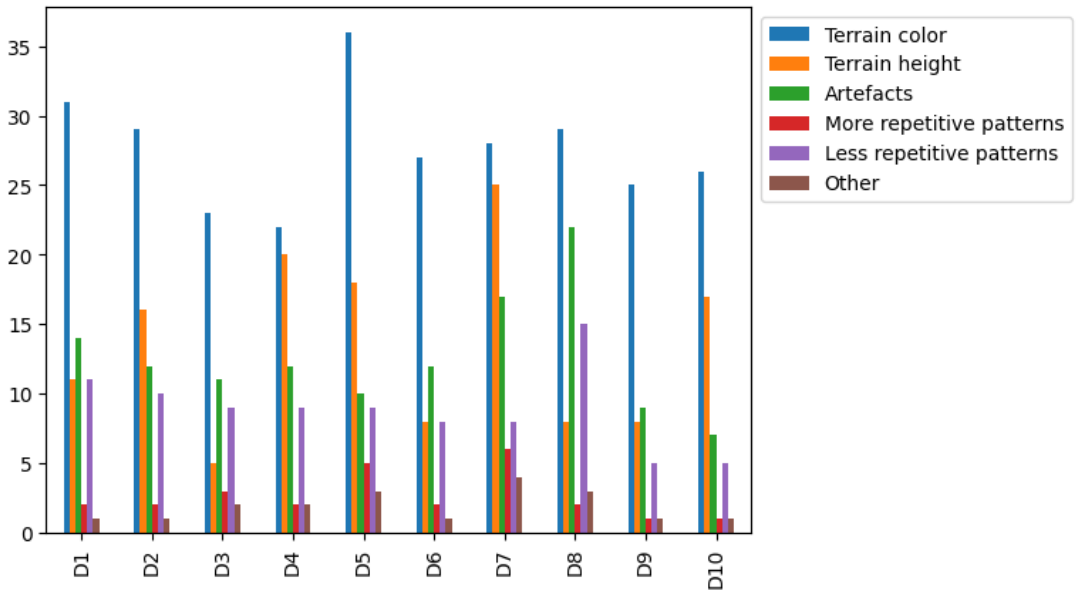


Figure 4.10: Frequency of reasons given for picking one terrain over another for generated desert terrains.

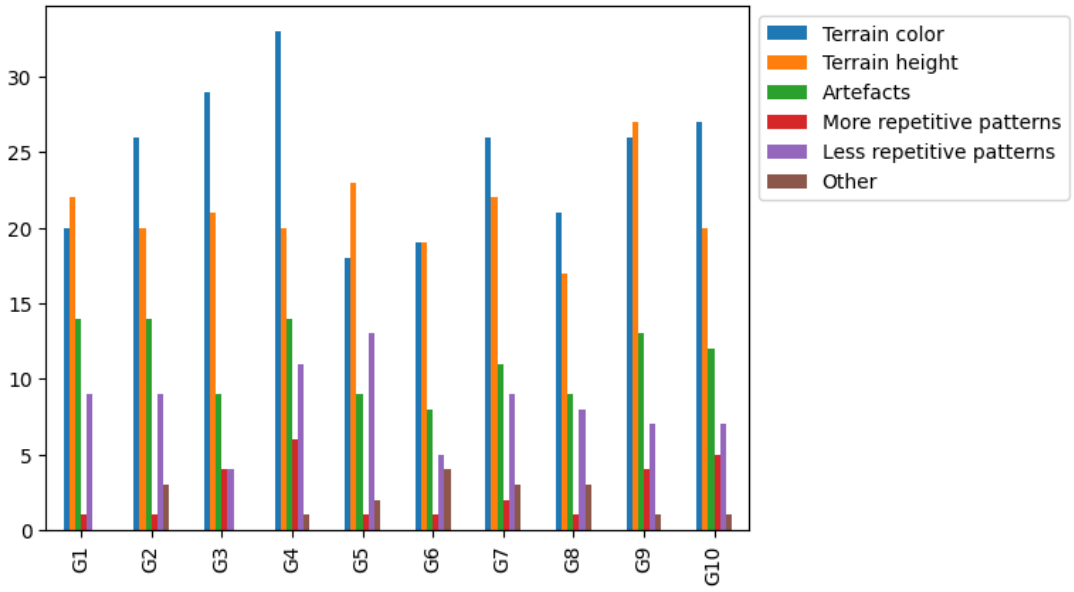


Figure 4.11: Frequency of reasons given for picking one terrain over another for generated Greece terrains.

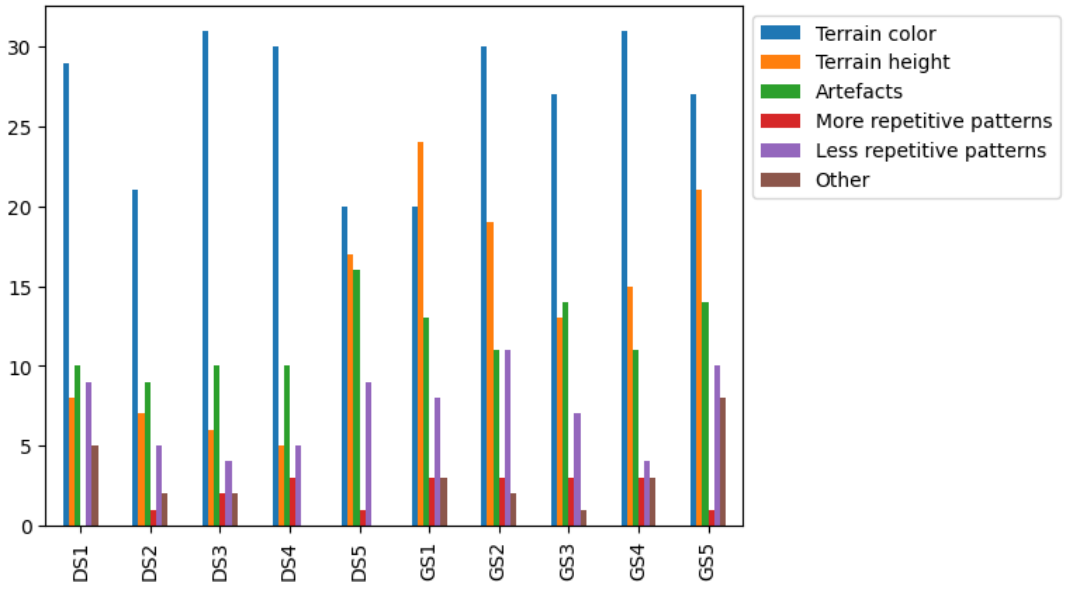


Figure 4.12: Frequency of reasons given for picking one terrain over another for comparisons of satellite images and generated terrains.

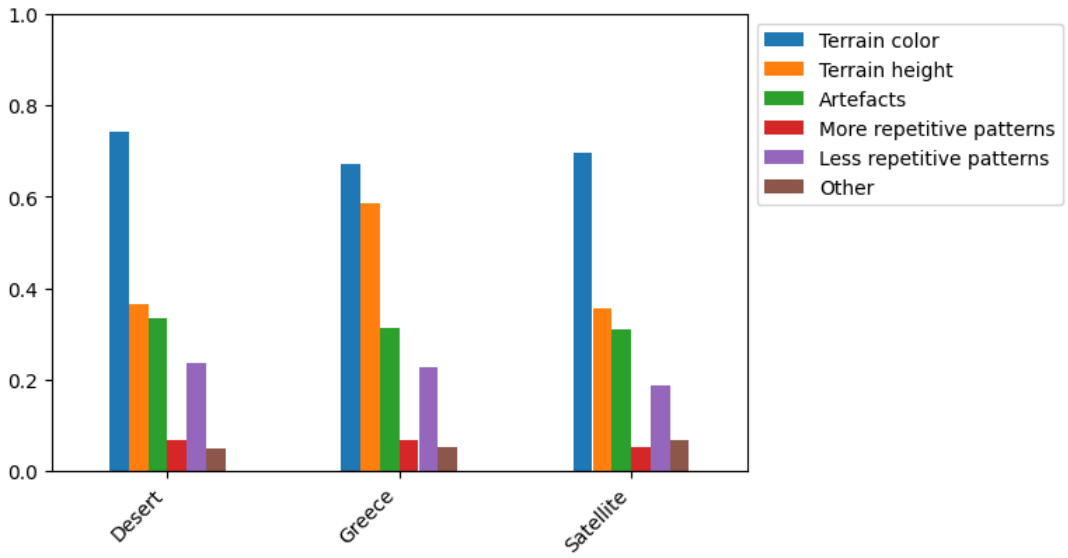


Figure 4.13: Mean average for each of the different comparison sets for the selected reasons.

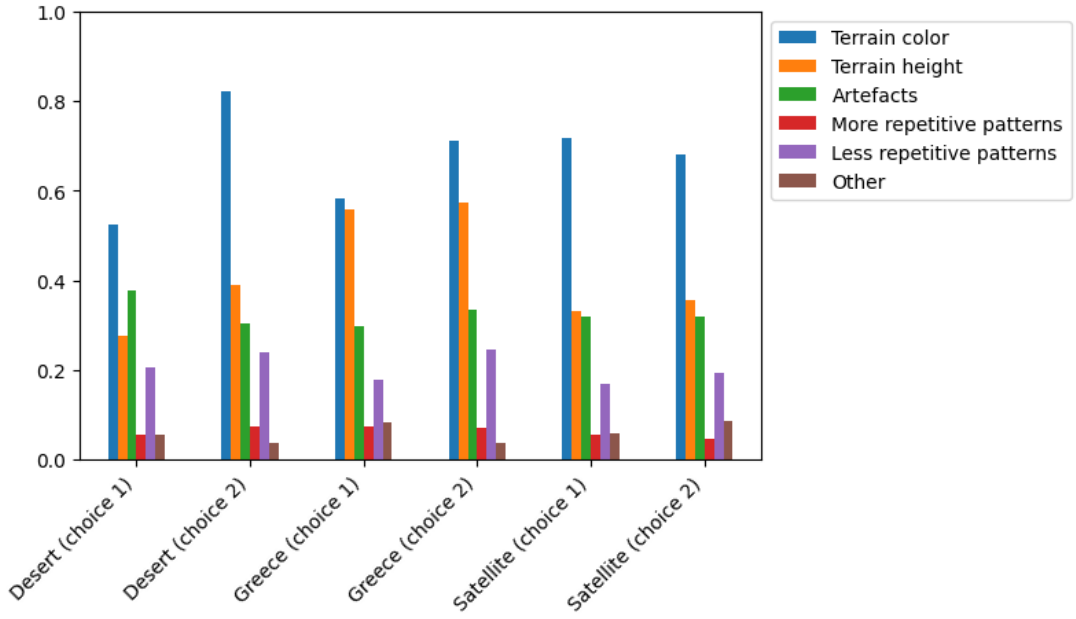


Figure 4.14: Mean average for each of the different comparison sets for the selected reasons, grouped by the selected choice.

From these graphs, it is apparent that terrain color was the most chosen reasoning for all the comparison sets. It is noteworthy that this reasoning was chosen much more often when the terrain generated by SingleTerrainGAN was selected in the desert comparison questions. The next most chosen reasoning is the terrain height reasoning, although this reasoning was chosen more often in the Greece comparison set in comparison to the other sets. The other reasonings seem to be chosen in the same manner across the different comparison sets.

For the “other” option, this field has been used for several reasons. Some participants used the field to indicate that both terrains look either good or bad, while some indicated a difference in detail for the chosen terrain. An overview of all filled in “other” values for each comparison can be found in appendix B. After each displayed reason the chosen option is also shown in between brackets, with “1” corresponding to terrains generated by the previous models or satellite images, and “2” corresponding to terrains generated by SingleTerrainGAN.

4.2.3 Correlations

During the survey, participants were asked whether they are familiar with the types of terrain that are being compared, the frequency of visiting countries that include these types of terrains, and whether they have any expertise in terrain design. These questions were grouped into separate categories, and was used to look for correlations between the answers to these questions and the options chosen during the comparison section of the survey.

Correlating the data is done by computing the Spearman rank-order correlation over the terrain experience questions, and the respective options that were chosen in the comparison section. A two-sided p-value is also computed to check whether the correlation is statistically significant.

A table containing the Pearson Correlations and two-sided p-values over the comparison questions and terrain experience data can be found in table 4.1 and table 4.2. These tables are split into two separate tables to account for the different terrain types. P-values lower than 0.05 are highlighted in bold, as this can indicate statistical significance.

		FAMILIARITY WITH DESERT LANDSCAPES	FREQUENCY OF VISITING COUNTRIES WITH DESERTS	EXPERTISE IN TERRAIN DESIGN
D1	PC	0.134	0.292	0.135
	p-value	0.434	0.083	0.433
D2	PC	0.118	-0.016	-0.067
	p-value	0.499	0.931	0.704
D3	PC	0.004	0.020	-0.013
	p-value	0.983	0.913	0.943
D4	PC	-0.006	0.147	**-0.394**
	p-value	0.971	0.422	**0.024**
D5	PC	0.122	0.092	0.201
	p-value	0.432	0.566	0.190
D6	PC	-0.037	**0.387**	0.077
	p-value	0.833	**0.024**	0.658
D7	PC	-0.185	0.146	**-0.354**
	p-value	0.229	0.355	**0.018**
D8	PC	-0.038	0.234	0.033
	p-value	0.806	0.136	0.831
D9	PC	0.004	0.246	-0.161
	p-value	0.983	0.167	0.362
D10	PC	0.165	**0.506**	-0.029
	p-value	0.367	**0.005**	0.877
DS1	PC	-0.050	-0.018	-0.088
	p-value	0.755	0.911	0.581
DS2	PC	-0.112	-0.113	-0.202
	p-value	0.503	0.504	0.224
DS3	PC	-0.169	-0.136	-0.153
	p-value	0.317	0.435	0.365
DS4	PC	-0.261	0.259	0.066
	p-value	0.113	0.133	0.692
DS5	PC	-0.139	-0.128	0.144
	p-value	0.413	0.451	0.394

Table 4.1: Spearman rank-order correlation and two-sided p-values over desert comparison questions and background information questions, with statistically significant values highlighted in bold.

		FAMILIARITY WITH MEDITERRANEAN LANDSCAPES	FREQUENCY OF VISITING MEDITERRANEAN COUNTRIES	EXPERTISE IN TERRAIN DESIGN
G1	PC	-0.298	0.196	0.311
	p-value	0.097	0.291	0.083
G2	PC	0.063	0.037	0.299
	p-value	0.723	0.840	0.086
G3	PC	-0.132	-0.138	0.175
	p-value	0.425	0.407	0.285
G4	PC	0.141	0.102	0.110
	p-value	0.349	0.511	0.469
G5	PC	0.070	**0.342**	0.038
	p-value	0.689	**0.048**	0.829
G6	PC	0.262	0.108	0.250
	p-value	0.141	0.564	0.161
G7	PC	-0.105	-0.115	0.108
	p-value	0.525	0.499	0.511
G8	PC	0.165	-0.188	0.106
	p-value	0.374	0.328	0.570
G9	PC	0.238	**0.450**	-0.075
	p-value	0.157	**0.005**	0.659
G10	PC	0.277	0.065	-0.154
	p-value	0.097	0.710	0.362
GS1	PC	-0.005	0.174	0.078
	p-value	0.974	0.316	0.646
GS2	PC	0.039	-0.090	-0.029
	p-value	0.810	0.587	0.856
GS3	PC	-0.133	-0.046	0.071
	p-value	0.406	0.783	0.659
GS4	PC	0.114	0.004	-0.127
	p-value	0.519	0.983	0.473
GS5	PC	-0.116	-0.083	0.024
	p-value	0.487	0.631	0.885

Table 4.2: Spearman rank-order correlation and two-sided p-values over Greece comparison questions and background information questions, with statistically significant values highlighted in bold.

From these tables, questions D4 and D7 have a p-value lower than 0.05 for participants that indicated to have expertise in terrain design. D4 has a Pearson Correlation of -0.394, while D7 has a Pearson Correlation of -0.354. Furthermore, question G5 has a p-value lower than 0.05 related to the frequency of visiting a country located in the Mediterranean area. The Pearson Correlation here is 0.425.

Apart from these correlations, a correlation between the chosen answer and the level of confidence the participants indicated to have when choosing these options can also be made. A table containing the Pearson Correlations and two-sided p-values of these can be seen in table 4.3. In this table, comparison questions D6, D10 and G9 have a p-value lower than 0.05. These correlations also reflect in figures 4.6 and figure 4.7, in which the confidence level for each choice is shown in a graph.

		CONFIDENCE LEVEL
D1	Correlation	0.282
	p-value	0.096
D2	Correlation	0.004
	p-value	0.981
D3	Correlation	0.011
	p-value	0.95
D4	Correlation	0.126
	p-value	0.479
D5	Correlation	0.095
	p-value	0.54
D6	Correlation	**0.371**
	p-value	**0.028**
D7	Correlation	0.175
	p-value	0.256
D8	Correlation	0.255
	p-value	0.091
D9	Correlation	0.231
	p-value	0.19
D10	Correlation	**0.428**
	p-value	**0.015**
DS1	Correlation	-0.053
	p-value	0.739
DS2	Correlation	-0.114
	p-value	0.496
DS3	Correlation	-0.128
	p-value	0.451
DS4	Correlation	0.257
	p-value	0.119
DS5	Correlation	-0.149
	p-value	0.378
G1	Correlation	0.145
	p-value	0.429
G2	Correlation	0.092
	p-value	0.605
G3	Correlation	-0.135
	p-value	0.412
G4	Correlation	0.139
	p-value	0.358
G5	Correlation	0.32
	p-value	0.061
G6	Correlation	0.13
	p-value	0.47

G7	Correlation	-0.069
	p-value	0.675
G8	Correlation	-0.147
	p-value	0.429
G9	Correlation	**0.45**
	p-value	**0.005**
G10	Correlation	0.049
	p-value	0.774
GS1	Correlation	0.235
	p-value	0.161
GS2	Correlation	-0.052
	p-value	0.749
GS3	Correlation	-0.017
	p-value	0.918
GS4	Correlation	0.015
	p-value	0.932
GS5	Correlation	-0.1
	p-value	0.549

Table 4.3: Spearman rank-order correlation and two-sided p-values over comparison questions and confidence level, with p-value lower than 0.05 highlighted in bold.

4.3 Convergence and FID

For the convergence of the models, as mentioned in chapter 3, the FID for intermediate models is plotted in a graph to see how the FID progresses during the training process. A lower FID means that the output generated by the GAN(s) is more similar to the original dataset.

The FID Score progression for Beckham and Pal's (2017) model and SingleTerrainGAN trained on the desert dataset can be seen in figure 4.15. The same thing for Panagiotou and Charou's (2020) model and SingleTerrainGAN trained on the Greece dataset can be seen in figure 4.16. These models are paired in one graph because both are trained on the same dataset. The graphs are plotted on different x-axes, as the models use different quantifications for training iteration.

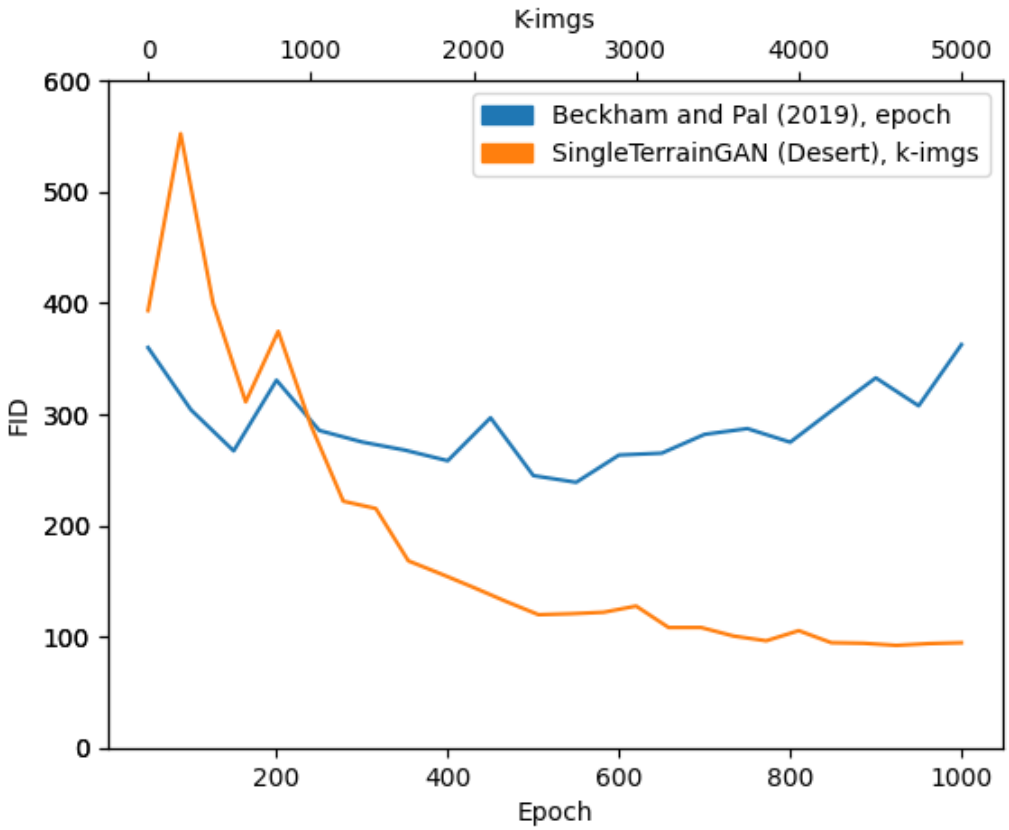


Figure 4.15: FID on intermediate models of Beckham and Pal's (2017) GANs and SingleTerrainGAN trained on the desert dataset.

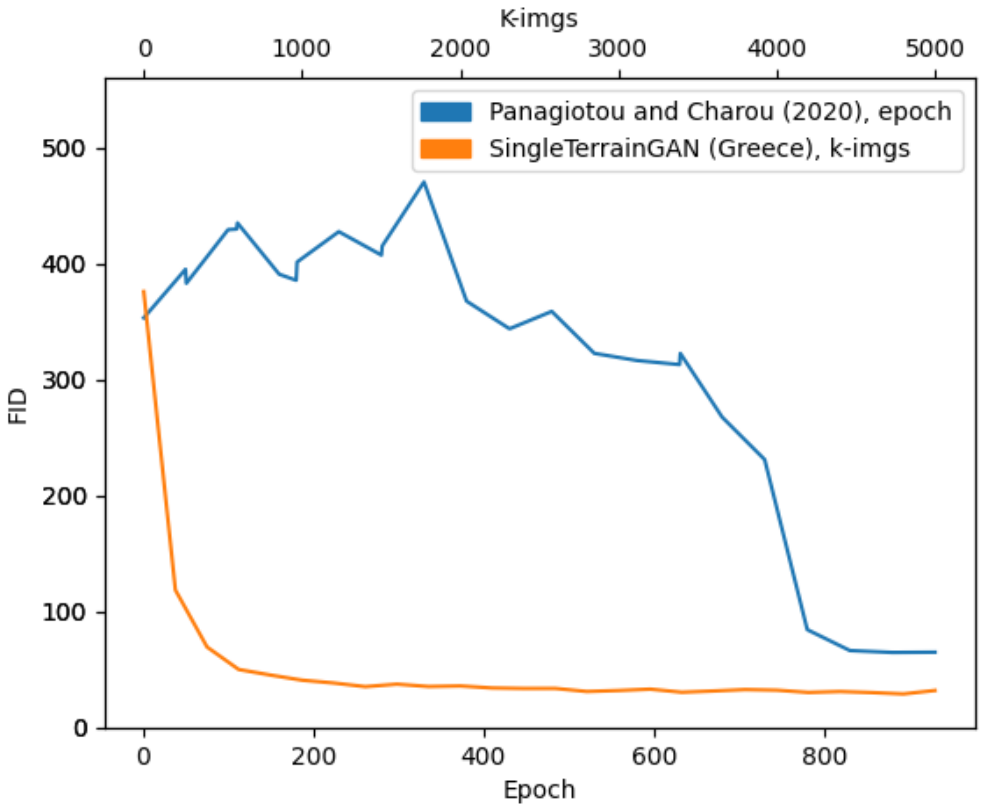


Figure 4.16: FID on intermediate models of Panagiotou and Charou's (2020) GANs and SingleTerrainGAN trained on the Greece dataset.

From these figures, it is apparent that the FID for Beckham and Pal's model drops down and increases irregularly as training progresses. The lowest FID Score for the intermediate models is at epoch 550, with an FID Score of 238.7. On the other hand, the FID Score of SingleTerrainGAN trained on the desert dataset decreases as training progresses, spiking up a few times at the start of the training, and eventually lowering to around 93. The lowest FID Score for this model is 92.1 at 4600 k-imgs.

In figure 4.8 it can also be seen that the FID for Panagiotou and Charou's (2020) model goes up and down during the training process, until eventually dropping down swiftly at the end of the training process. The lowest FID on an intermediate model was found for epoch 880, with an FID of 64.6. In contrast, the intermediate models of SingleTerrainGAN display a different behaviour where their FID values rapidly decrease at the beginning of the training process, stabilizing around an FID of 30. The lowest FID found during this training process is 28.8 at 4800 k-imgs.

Because SingleTerrainGAN's FID score seems to flatten at around 3000 k-imgs on the desert-trained model, and at 1000 k-imgs on the Greece-trained model, we zoom into these graphs in figure 4.17 and figure 4.18.

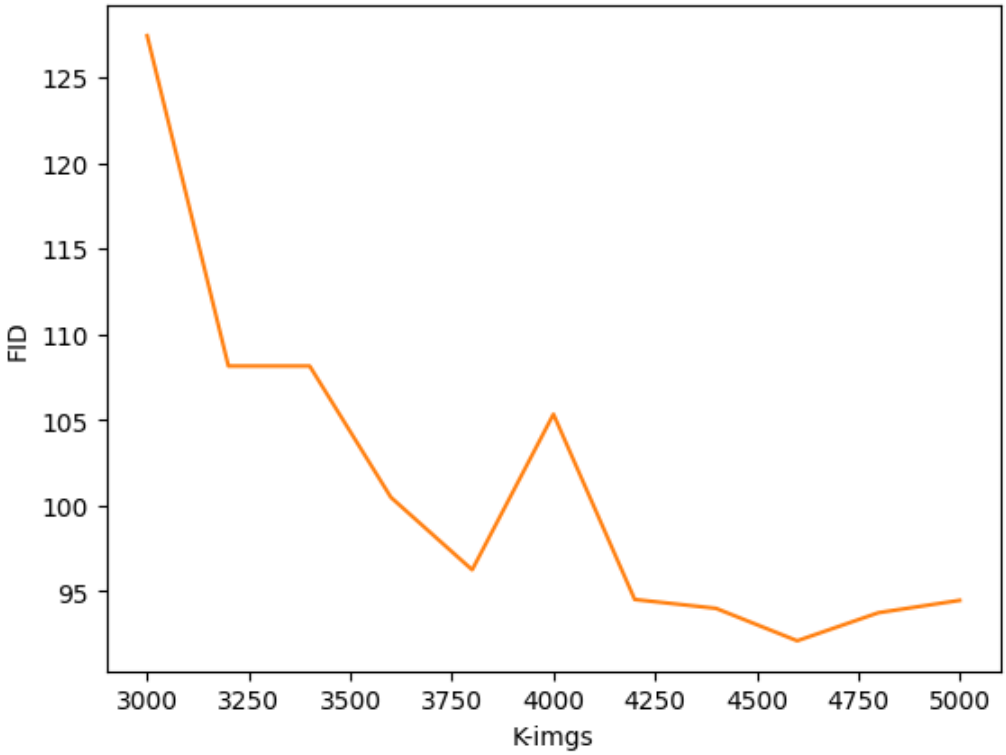


Figure 4.17: FID on intermediate models of SingleTerrainGAN trained on the desert dataset, from k-imgs 3000 to 5000.

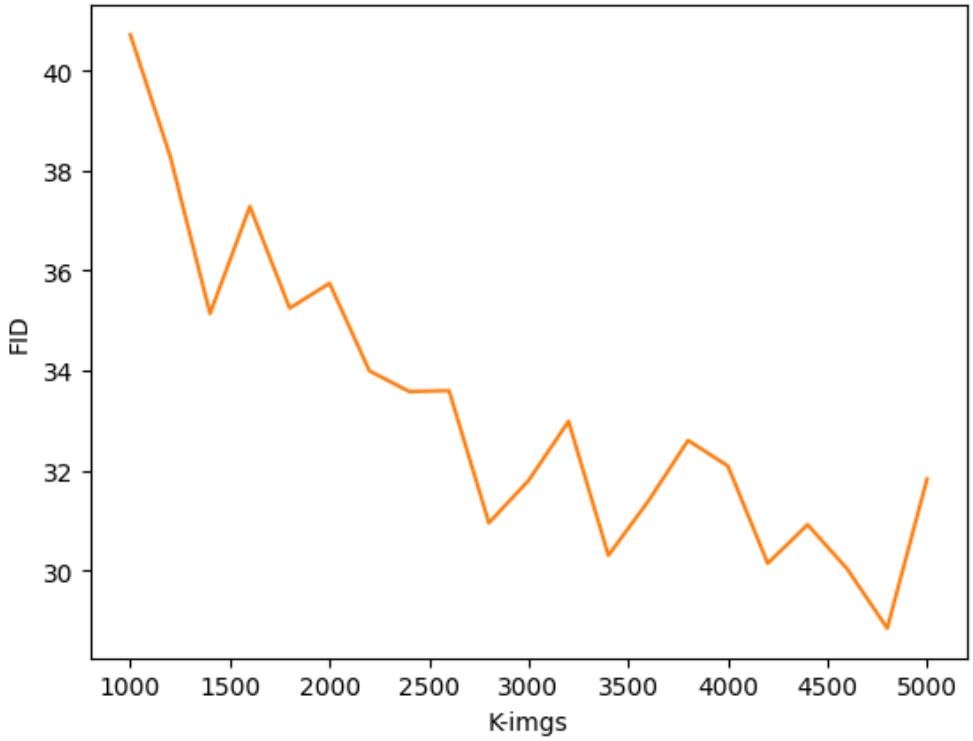


Figure 4.18: FID on intermediate models of SingleTerrainGAN trained on the Greece dataset, from k-imgs 1000 to 5000.

From these figures, we can see that the FID score trend of both trained models is declining slowly during the rest of the training process.

4.4 Training time and Computational performance

The training time, mean generation time and peak GPU memory allocated during generation can be seen in table 4.4 and table 4.5. These tables are purposely split up into two to make it easier to compare SingleTerrainGAN trained with the same dataset as one of the earlier models.

	BECKHAM AND PAL (2017)	**SINGLETERRAINGAN (DESERT)**
TRAINING TIME	6 hours and 31 minutes	4 days, 1 hour and 56 minutes
MEAN GENERATION TIME	15.12 milliseconds	17.83 milliseconds
PEAK GPU MEMORY ALLOCATED	17547 MiB	2294 MiB

Table 4.4: Training time, mean generation time and peak GPU memory allocation for Beckham and Pal's (2017) GANs and SingleTerrainGAN trained on the desert dataset.

	PANAGIOTOU AND CHAROU (2020)	**SINGLETERRAINGAN (GREECE)**
TRAINING TIME	1 day, 8 hours and 44 minutes	1 day, 5 hours and 11 minutes
MEAN GENERATION TIME	10.53 milliseconds	12.78 milliseconds
PEAK GPU MEMORY ALLOCATED	14785 MiB	2096 MiB

Table 4.5: Training time, mean generation time and peak GPU memory allocation for Panagiotou and Charou's (2020) GANs and SingleTerrainGAN trained on the Greece dataset.

In table 4.1, it is apparent that SingleTerrainGAN trained on the desert dataset has a much longer training time than the GANs by Beckham and Pal (2017). The peak GPU memory that is allocated during the generation is much lower, with a difference of 15253 MiB (~16 GB). The experiment revealed that TensorFlow models consumed significantly more GPU memory compared to PyTorch models, even with memory growth turned on. On the other hand, the mean generation time for SingleTerrainGAN is slightly higher.

In table 4.2, SingleTerrainGAN trained on the Greece dataset has a slightly lower training time compared to the GAN by Panagiotou and Charou (2020). As before, the peak GPU memory allocated during generation was a lot lower using SingleTerrainGAN, with a difference of 12689 MiB (~13.3 GB). The mean generation time for SingleTerrainGAN trained on the Greece dataset is again slightly higher than the GANs it is being compared to.

Chapter 5: Discussion of Data and Results

At the end of chapter 2, a research question for this thesis was declared as follows: what are the effects of using a single GAN to generate both a heightmap and texture for terrain compared to using two separate GANs, in terms of realism, convergence, training time, and computational performance? The data and results stated in chapter 4 set out to give an answer to this question. This chapter will go over each metric derived from the research question one-by-one to interpret the data, and discuss what this means for each individual metric.

5.1 Realism

As mentioned in chapter 3, realism is evaluated by looking at both the survey, and the FID for the models and comparing these values.

5.1.1 Survey

Looking at the survey results, it can be seen by examining the p-values obtained from the chi-square tests that the values are all below 0.01. This indicates that the null hypothesis can be rejected, which implies that there is a meaningful difference in perceived realism for all comparison sets.

In figure 4.2 it is evident that terrains generated by SingleTerrainGAN are picked more often in all comparison questions, meaning that participants find the terrains generated by SingleTerrainGAN more realistic as opposed to terrains generated by Beckham and Pal's (2017) model.

For the terrain comparison questions for terrains generated from the Greece dataset, we can see that SingleTerrainGAN has also been chosen more often in most cases, with some comparisons being more even, and one outlier where the terrain generated by Panagiotou and Charou's (2020) model appeared more realistic to the participants. With a mean pick rate of 0.637, we can still argue that SingleTerrainGAN generates more realistic-looking terrain in general compared to Panagiotou and Charou's model.

If we look at the terrain comparison questions between rendered satellite images and generated terrains in figure 4.3, we can see that rendered satellite images still outperform SingleTerrainGAN, having a mean pick rate of 0.327.

The mean confidence level plotted in 4.8 shows a slight drop during the second half of the survey. This could indicate that participants felt more unconfident as the survey progressed, although the difference in confidence between the start and the end of the survey is only less than half a point. Another reason could be that the default value on the confidence slider was 1, and some participants did not bother to change this value during the second half of the survey. Figure 4.9 does seem to support this reasoning, as a confidence level of 1 was chosen more often during the second half of the survey, although not by a lot.

When looking at the participants' reasoning for choosing one terrain over another, we can see in figure 4.12 that participants mostly look at the terrain colour and the terrain height as their reasoning to choose one terrain over another. The terrain height has been consistently selected more frequently in the Greece dataset, likely since the desert dataset lacks variations in height in most cases. Participants also seem to look at visual artefacts whenever these are present in the terrain, looking at figure 4.9.

Looking at the correlations between terrain familiarity or terrain design experience, there are a few questions where there is some statistical significance ($p \leq 0.05$) and a moderate correlation ($0.3 \geq |PC| \geq 0.5$). For the desert terrain comparison questions, participants with terrain design experience chose the terrain not generated by SingleTerrainGAN more often whenever there were visible artefacts present in the terrain generated by SingleTerrainGAN, compared to participants with no terrain design experience.

There is also a moderate correlation between the indicated frequency of visiting Mediterranean countries and comparison G5, in which these participants chose the terrain generated by SingleTerrainGAN more often.

Because there were only a few correlations found for a few questions, it can be argued that in general terrain familiarity and terrain design experience do not correlate with the options that are picked in the comparison questions.

5.1.2 FID

The FID is a metric score that can be used to see how similar a generated dataset is to the initial dataset. A lower score means that the generated dataset is more similar to the initial dataset, while a higher score means that the generated dataset is less similar to

the initial dataset. As the initial dataset is the most realistic the generated terrains can get, we can argue that a lower FID also means more realistic looking terrains are output by the generator.

The FID for each model can be derived from the graphs pictured in figure 4.7 and figure 4.8. The lowest points in these graphs are taken to compare them against each other.

The lowest point for Beckham and Pal's (2017) model is 238.7 at epoch 550. We can compare that to the lowest point for SingleTerrainGAN, which is 92.1 at 4600 k-imgs. This means that the FID for SingleTerrainGAN is much lower than the FID for Beckham and Pal's model. Thus, we can argue that SingleTerrainGAN produces more realistic looking terrains than Beckham and Pal's model when looking at the FID for both models.

The lowest FID point for Panagiotou and Charou's (2020) model is 64.6, at epoch 880. SingleTerrainGAN has a lowest FID point of 28.8 at 4800 k-imgs. While the absolute difference between these values is not as big as the one that can be seen between Beckham and Pal's model and SingleTerrainGAN, the terrains generated by SingleTerrainGAN still seem to be much more similar to the initial dataset according to the FID. Subsequently, we can also argue that SingleTerrainGAN produces more realistic results than Panagiotou and Charou's model when looking at the FID for both models.

Looking at both the survey data and the FID, we can argue that these results agree with each other, in a way that SingleTerrainGAN does produce more realistic results in comparison to earlier models.

5.2 Convergence

As for convergence, we can again look at figure 4.7 and figure 4.8 to compare convergence between the models used in earlier studies and SingleTerrainGAN. From these figures, we can argue that SingleTerrainGAN converges much faster than the earlier models, while SingleTerrainGAN trained on the desert dataset does converge slower than SingleTerrainGAN trained on the Greece dataset. This is most likely because the desert dataset uses images of a higher resolution.

For Beckham and Pal's (2017) model, the model itself does not seem to be converging throughout the training process. This could be because the model itself was recreated completely in TensorFlow, meaning that the output generated by this recreated GAN is

different from the initially developed GAN. Comparing our results to the results that are shown in Beckham and Pal's (2017) paper, we can see that the output generated by both models looks very similar, which could mean that the initial GAN did not perform that well.

5.3 Training time

The training time for each model can be seen in table 4.1 and table 4.2. Using these tables, we can see that the training time for Beckham and Pal's (2017) is only 6 hours and 31 minutes, while the training time for SingleTerrainGAN trained on the desert dataset is 4 days, 1 hour and 56 minutes. This is mostly because Beckham and Pal used a much simpler model, as compared to the StyleGAN2 model SingleTerrainGAN uses.

The training time between Panagiotou and Charou's (2020) and SingleTerrainGAN trained on the Greece dataset do not differ much, with Panagiotou and Charou's model even training around 3.5 hours longer.

The difference between the training time of SingleTerrainGAN trained on the desert dataset, and SingleTerrainGAN trained on the Greece dataset, can be explained by the resolution difference of the datasets. The desert dataset contains satellite images with a resolution of 512x512 pixels, while the Greece dataset contains satellite images with a resolution of 256x256 pixels.

5.4 Computational performance

Computational performance includes both the generation time and memory footprint for each model, which can also be seen in table 4.1 and table 4.2.

The mean generation times for terrains generated by SingleTerrainGAN are slightly worse than the models they are being compared to, although they are only generating for 2 milliseconds longer on average.

On the contrary, SingleTerrainGAN allocates much less GPU memory compared to the models they compare to. This is very important for ensuring the usability of SingleTerrainGAN on systems with limited GPU memory.

5.5 Limitations in research

Although the results seem conclusive, there are some limitations that can be found in this research.

Firstly, a limitation in the research is that the GANs from previous studies were either recreated in the case of Beckham and Pal's (2017) model, or used a slightly different version of the model for training in the case of Panagiotou and Charou's (2020) model. This means that it could be the case that the output given by these models could be slightly different than observed during the initial studies.

A second limitation can be found in the training process of SingleTerrainGAN. The model is trained on 5.000 k-imgs, while the model could benefit from training for longer as the FID seems to be decreasing near the end of the training process, albeit very slowly. Although the FID for SingleTerrainGAN is much lower than earlier models, the FID could reach an even lower point when training for longer, thus possibly improving the realistic appearance of the generated terrains.

Another limitation can be found in the survey, for which several terrains were compared against each other. The samples used for these comparisons were manually picked. This means that it is possible that, while it appeared that SingleTerrainGAN produces more realistic results, bad samples for the models used in previous studies were chosen. This could lead to some unfair comparisons, in which participants are more likely to choose terrains generated by SingleTerrainGAN. A possible solution for future studies is that these terrains could be algorithmically picked in terms of similarity, thus preventing this problem.

A limitation also related to the survey, is the number of participants that filled in the survey. As the number of participants that fully filled in the survey is only 51, and partially filled in is 23, the survey results could be of more significance if more people filled in the survey.

An additional limitation can be found in the visualization of the terrains. The terrains showcased in the survey are shown from a distance, which can give an entirely different look as opposed to seeing the terrain up close. If the survey did force participants to see the terrains up close, the results might be different.

A last limitation can also be found in the training time metric of the GANs used for the study by Beckham and Pal (2017). In table 4.1 this model has a training time of 6 hours and 31 minutes, training for a total of 1000 epochs, while the lowest FID found in this model is seen at epoch 550. This means that this model has trained for nearly double the number of epochs than necessary, leading to a higher training time.

Chapter 6: Conclusion and Future Directions

6.1 Conclusion

This thesis aims to answer the following question: what are the effects of using a single GAN to generate both a heightmap and texture for terrain compared to using two separate GANs, in terms of realism, convergence, training time, and computational performance?

From the data and results shown in chapter 4, and after discussing these results in chapter 5, we can conclude that the single GAN developed for this paper, now known as SingleTerrainGAN, does outperform the earlier developed GANs by Beckham and Pal (2017) and Panagiotou and Charou (2020) in terms of realism and convergence.

Both the survey and the FID scores the GANs delivered agree that SingleTerrainGAN produces more realistic results, with SingleTerrainGAN being chosen over the earlier models in the survey for 85% of the cases, and the FID score decreasing by over 50%.

SingleTerrainGAN also starts to converge much faster compared to earlier models, as it already starts converging at the start of the training process. Earlier models started converging much later in the training process, or did not converge at all in the case of Beckham and Pal's (2017) model.

As for training time, we can conclude that SingleTerrainGAN either trains a lot longer when compared to a simpler model, as is the case when compared with Beckham and Pal's model, or trains in a similar time frame when training against a model with more trainable variables such as Panagiotou and Charou's model. However, this does not hold up when also taking the intermediate FID scores into account, as SingleTerrainGAN outperformed the earlier models early on in the training process, meaning that training could have been stopped earlier.

Concerning computational performance, we can conclude that generating using SingleTerrainGAN was slightly slower compared to using two separate GANs, although only a difference of 2 milliseconds was found, giving a generation time of 17.8 milliseconds and 12.8 milliseconds for SingleTerrainGAN trained on the different datasets. The memory footprint, on the other hand, is lowered by a huge margin, with a

difference of 16 gigabytes in comparison to Beckham and Pal's (2017) model, and a difference of 13.3 gigabytes in comparison to Panagiotou and Charou's (2020) model.

6.2 Future directions

As deep learning is a fast-growing field, there is a lot more to analyse in future research endeavours. One option is to train the GAN on a larger variety of different biomes at the same time, to see whether the GAN generates a terrain in which the biomes are combined, or whether the GAN generates terrains that consist of a single biome. This would require creating a new dataset that contains satellite images and heightmaps of these biomes.

As the datasets used to train SingleTerrainGAN contains only small-sized images, the lack of quality can be seen when viewing the terrains up close. This also means that the terrains are not fit for video games in cases where the player must view the terrain up close, such as a game where the player is playing from a first-person perspective. It would be interesting to see what happens when the GAN is trained on a dataset with upscaled textures and heightmaps, or a dataset with higher resolution images. This would lead to a vastly different outcome in terms of quality, training time, convergence, and computational performance.

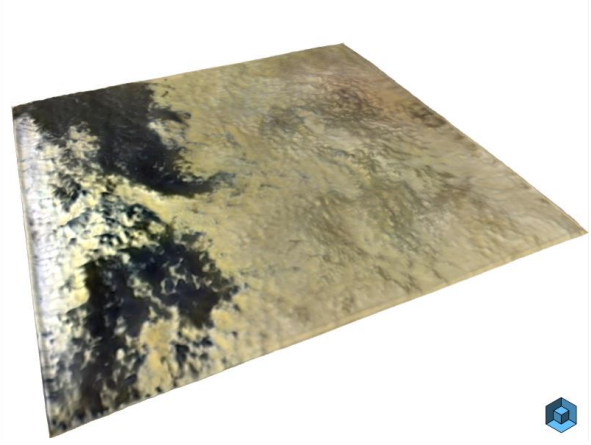
A recent development in the domain of deep learning is stable diffusion. Stable diffusion makes use of a diffusion model, which works by adding noise to the training data, and then reversing this process to recover the initial data during the learning process. This eventually leads the model to be able to generate images from noise. Stable diffusion can generate images by text prompts, as well as make image-to-image translations using text prompts. Thus, a texture can be generated using stable diffusion, and translated to a heightmap afterwards. This would allow for more control over the generated output, which can be desirable for the games industry. It would be interesting to see how the results stated in this paper hold up against terrains that could be generated by such models.

Appendix A: Overview of terrain comparison questions

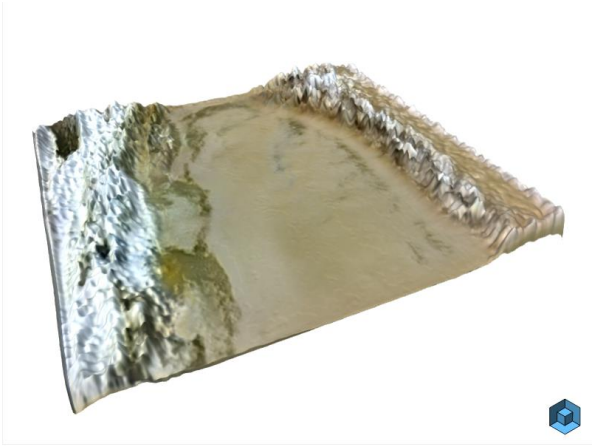
D1

D2

D3



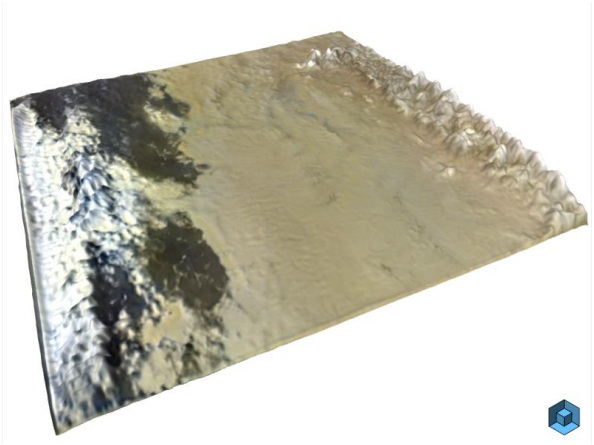
D4



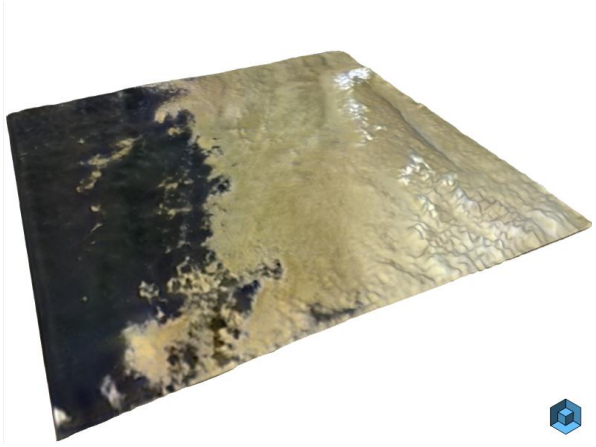
D5

D6

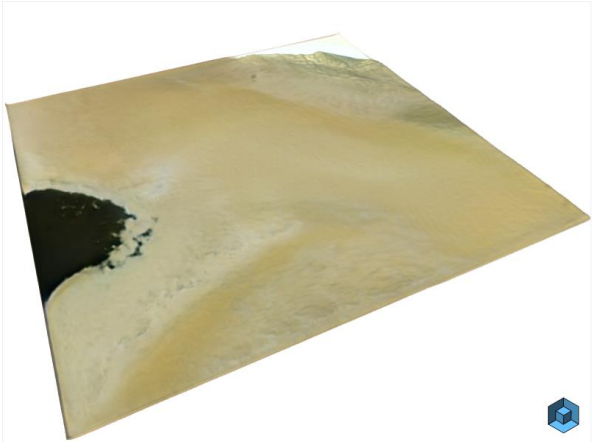
D7



D8



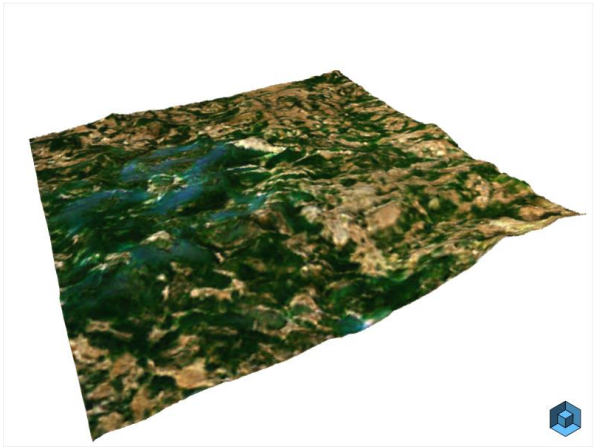
D9

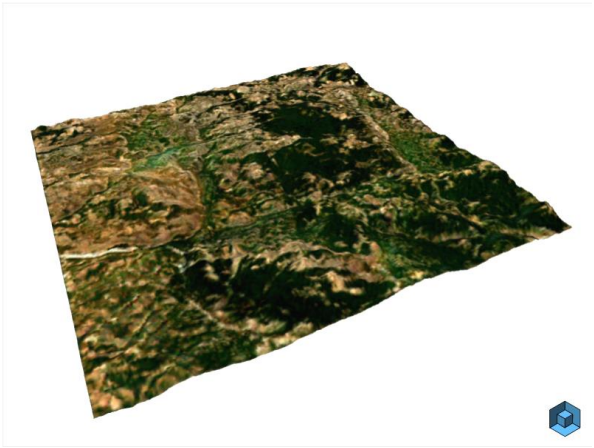


D10

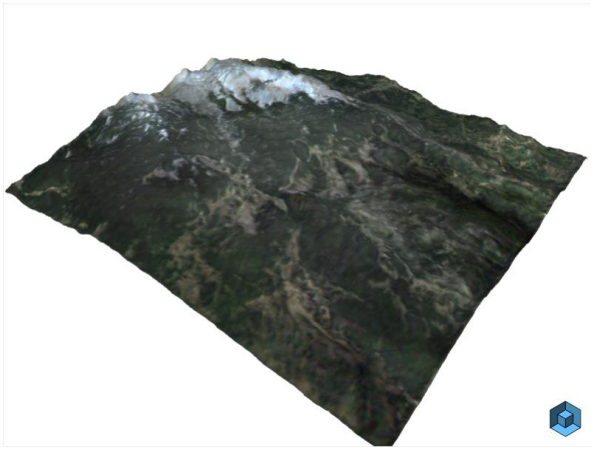
G1

G2





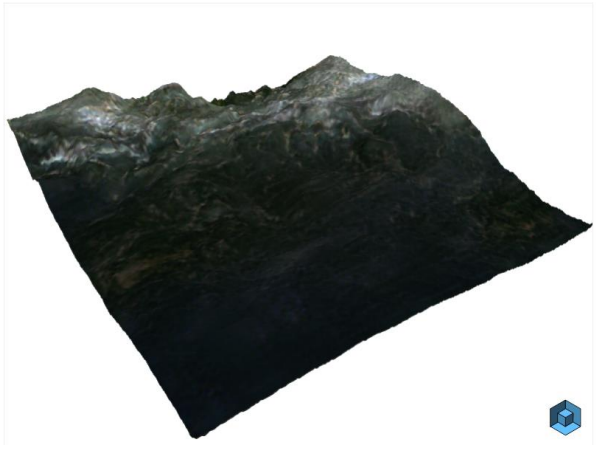
G3

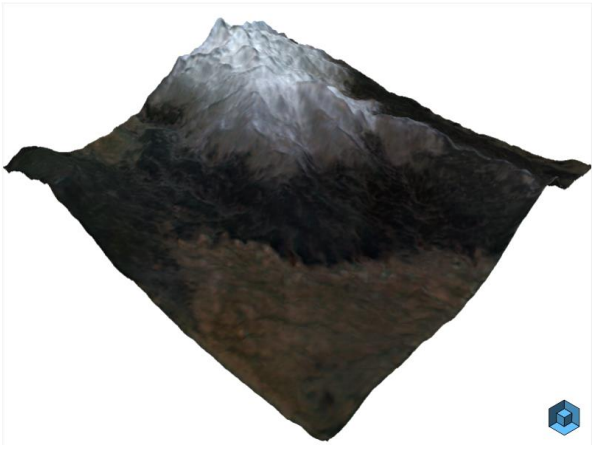


G4

G5

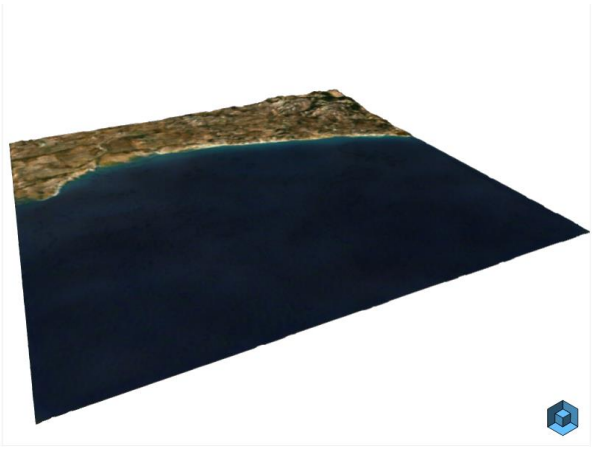
G6





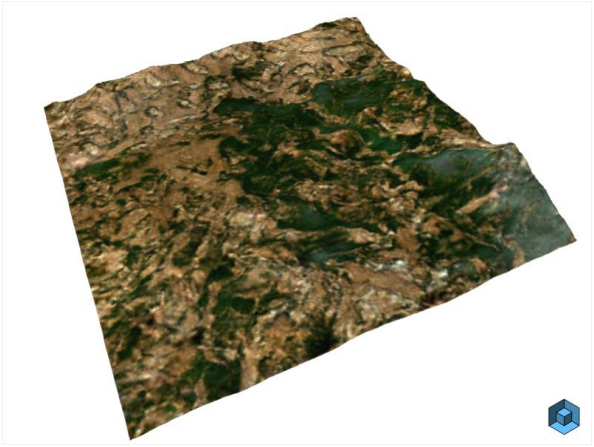
G7

G8

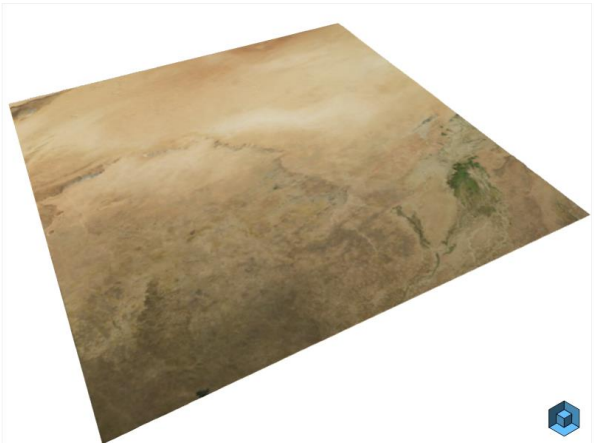


G9

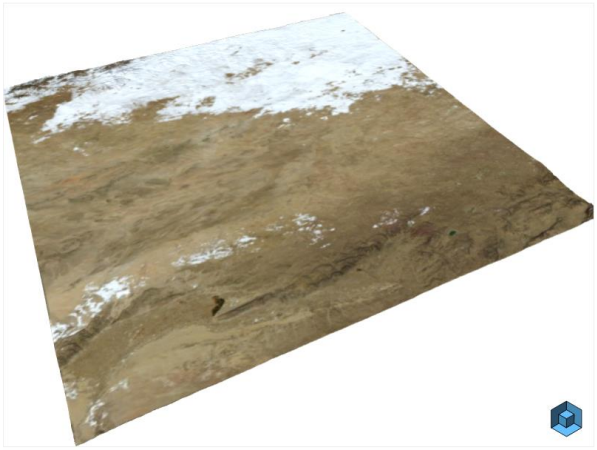
G10

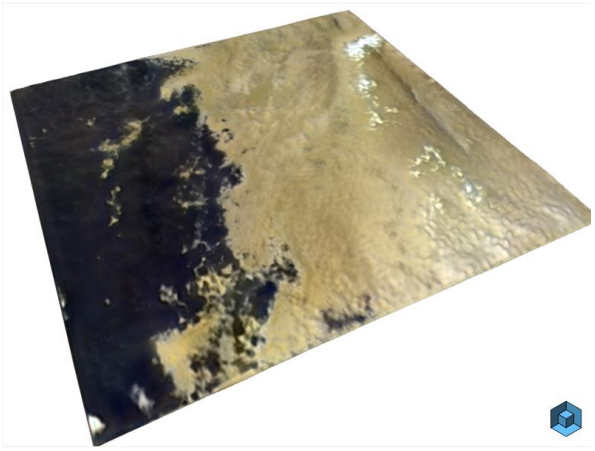


DS1

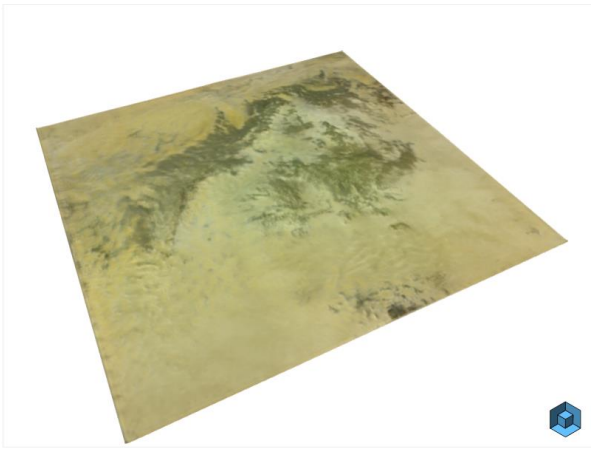


DS2



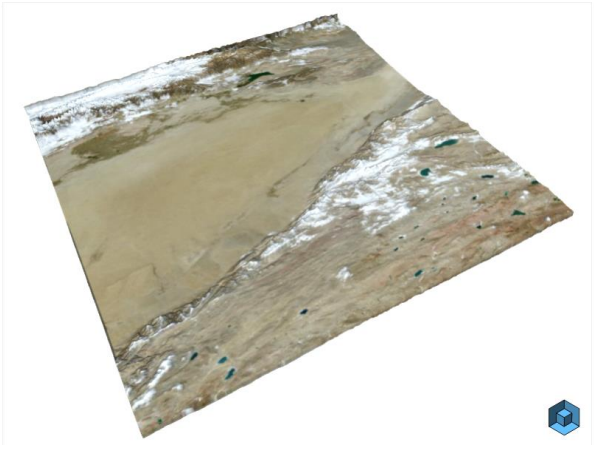


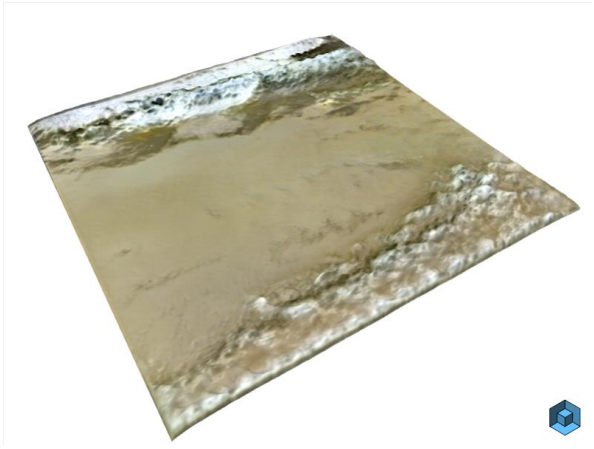
DS3



DS4

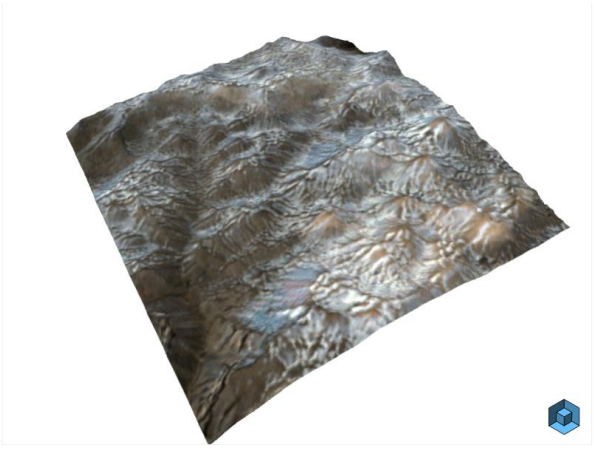
DS5

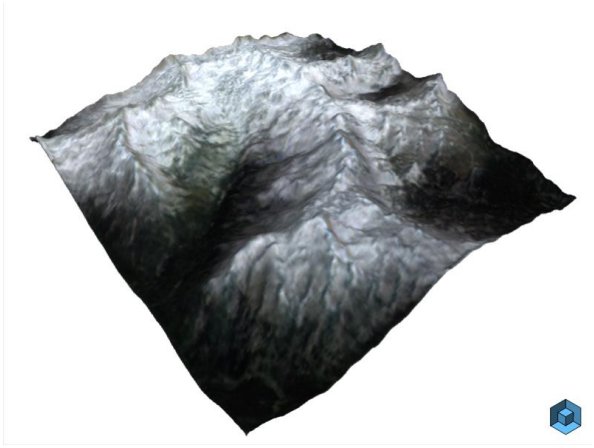




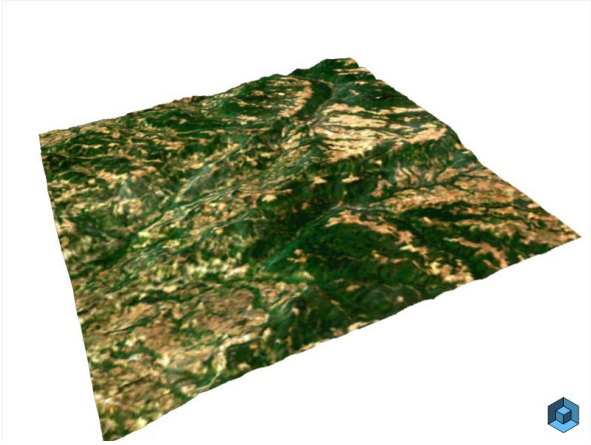
GS1

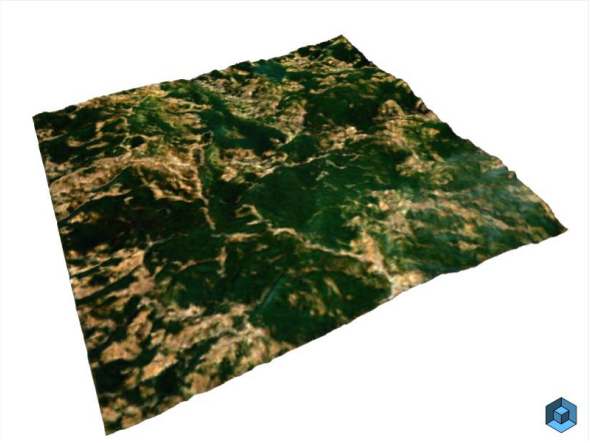
GS2



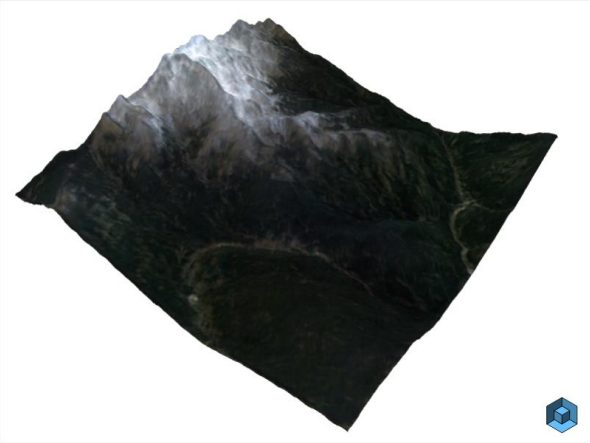


GS3

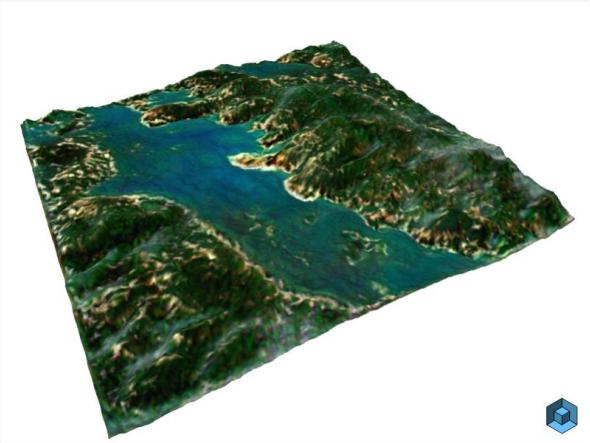




GS4



GS5



Appendix B: “Other” reasoning values

QUESTION	REASONING
D1	Other terrain has plaid patterns in the flat areas (2)
D2	This one is prettier (2)
D3	both not generated proper (2) More detail (2)
D4	Mountain ranges are too unnatural in the other one. (1) The terrain in the other one was way too inconsistent (1)
D5	both flat and not generated proper (2) Both not realistic (2) I can't see the other terrain. (2)
D6	They both don't look very realistic, but the other one's color is just wrong (1)
D7	They both look fake af (and also the one I chose has less blur) (1) more relief (2) Mountains are unnatural in the other one (1) The terrain I did not choose has very linear transitions which appear to me unnatural, although the resolution is very low and hard to decipher (1)
D8	both are bad (1) I can't see the other terrain but the left one has a grid like pattern on it. (2) None of them are as confinsing because of how glossy the terrain looks. (2)
D9	both are bad (2)
D10	The terrain I chose has better variety (2)
DS1	both are bad (1) The brighter (white) part is much more convincing (2) Less blurry (1) Just looks more authentic (1) both good (1)

DS2	both are bad (2) I can't see the other terrain. (2)
DS3	both are bad (2) both good (1)
DS4	-
DS5	-
G1	-
G2	Less blurry (2) Water distribution in the other terrain looks off (2) betere kleuren (better colors in Dutch) (1)
G3	-
G4	Water pathes from the other option look off (1)
G5	The terrain I chose has less blur and didn't look as much "random" as the other one (2) both good (1)
G6	The terrain I chose feels more natural (1) fewer snow (1) the terrain i chose has better blending areas, in the other the plains/forest/mountain look too sharply separated; also the terrain i chose looks more like what i am used to seeing on google maps if it makes any sense (1) Snow distribution seems more natural than the other one (2)
G7	More detail (2) the terrain i DIDN'T choose has a weird texture on the water but the overall shape of the coast looks better (2)
G8	The one I chose looks like it has more shadow (2) It looks very realistic with right proportion of the mountains, sea line and water. (1) both good (1)
G9	More detail (2)
G10	fewer forests (1)

GS1	Other one is very blurry/pixelated (1) more buldings (2) The terrain I picked feels more accurate in terms of perceived scale vs level of detail. The terrain I picked looks like a closer up scan of a rocky beach. The other one seems like a satellite view scan of the sea area of a country; and so the "expected" details and landmarks (swathe of beach, islands nearby, etc) aren't there. (1)
GS2	white/ snow till low in other example (2) the one shows just bigger area of the other one (2)
GS3	The higher the terrain, the less green it is. (1)
GS4	The terrain I chose has more natural feeling patterns (2) snow level of alternative seems different for peaks (1) both good (2)
GS5	The water is much more realistic (1) The terrain I chose abides to the laws of physics (water level doesn't differ as weirdly as the other example.) (1) Again: blur (1) The water layout and lack of weird peninsulas feels more natural (2) Terrain I chose has certain organic patterns that I would expect to see. Also, other terrain is confusing to me in terms of how I should locate it in geography. Is two tributaries merging into a river? Is it near the sea? The water edge seems too rough for either. (2) other has an odd level of beach vs high green area (1) The river is more natural (1) Other terrain has water height artifacts (1)

Bibliography

Mojang Studios (2009). *Minecraft*. Mojang Studios.

Hello Games (2016). *No Man's Sky*. Hello Games.

Blizzard North (1997). *Diablo*. Blizzard Entertainment.

McMillen, E. & Himsl, F. (2011). *The Binding of Isaac*. McMillen, E.

Liu, J., Snodgrass, S., Khalifa, A., Risi, S., Yannakakis, G. N., & Togelius, J. (2020). *Deep Learning for Procedural Content Generation*. https://doi.org/10.1007/s00521-020-05383-8

Beckham, C., & Pal, C. (2017). *A step towards procedural terrain generation with GANs*. http://arxiv.org/abs/1707.03383

Panagiotou, E., & Charou, E. (2020). *Procedural 3D Terrain Generation using Generative Adversarial Networks*. http://arxiv.org/abs/2010.06411

Cook, M. (2017). Ethical Procedural Generation. In *Procedural generation in Game Design*. essay, CRC Press, Taylor & Francis Group.

Shaker, N., Togelius, J., & Nelson, M. J. (2016). *Procedural Content Generation in Games*. Springer Publishing.

Smith, G. (2015). *An Analog History of Procedural Content Generation*.

Hendrikx, M., Meijer, S., van der Velden, J., & Iosup, A. (2013). Procedural content generation for games: A survey. *ACM Transactions on Multimedia Computing, Communications and Applications*, *9*(1). https://doi.org/10.1145/2422956.2422957

Kelleher, J. D. (2019, September 10). *Deep Learning (The MIT Press Essential Knowledge series)* (Illustrated). The MIT Press.

Melnychuk, V. (2020). *Landscape generation using procedural generation techniques*.

Brownlee, J. (2019). A gentle introduction to generative adversarial networks (GANs). *Machine Learning Mastery*, *17*.

Lopez-Garcia, E. (2019). *Deep Convolutional GANs for Real-Time Procedural Terrain Generation Systems*.

Naik, S., Jain, A., Sharma, A., & Rajan, K. S. (2022, July). Deep Generative Framework for Interactive 3D Terrain Authoring and Manipulation. In *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium* (pp. 6410-6413). IEEE.

Voulgaris, G., Mademlis, I., & Pitas, I. (2021). *Procedural Terrain Generation Using Generative Adversarial Networks*.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., & Aila, T. (2020). Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 8110-8119).

Karras, T., Laine, S., & Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4401-4410).

Borji, A. (2018). *Pros and Cons of GAN Evaluation Measures.* https://arxiv.org/pdf/1802.03446.pdf

Borji, A. (2022). *Pros and cons of GAN evaluation measures: New developments.* https://doi.org/10.1016/j.cviu.2021.103329

Slangewal, B. (2019). *Comparing Quantitative Metrics for Generative Adversarial Neural Networks*. TU Delft.

Xu, Q., Huang, G., Yuan, Y., Guo, C., Sun, Y., Wu, F., & Weinberger, K. Q. (2018). *An empirical study on evaluation metrics of generative adversarial networks*. https://arxiv.org/abs/1806.07755

This template is based on a template by:
Steve Gunn (http://users.ecs.soton.ac.uk/srg/softwaretools/document/templates/)
Sunil Patel (http://www.sunilpatel.co.uk/thesis-template/)

Template license:
CC BY-NC-SA 3.0 (http://creativecommons.org/licenses/by-nc-sa/3.0/)