



Microbial Content Generation for Natural Terrains in Computer Games

Maarten H. Lamers* and Wim J. O. M. van Eck

Leiden Institute of Advanced Computer Science, Leiden University, Leiden, Netherlands

Procedural content generation (PCG) has been applied since several decades to fulfill various game-related design needs. Besides bio-inspired methods, living (non-human) organisms were used in computer games for various purposes, such as behavior generation, data gathering, and player education. No living organisms were used for the generation of virtual terrains in games. Such an approach to terrain generation could benefit from morphological similarity between natural terrains and colonies of microbial organisms, real-time development of terrains over time, and educational opportunities. We successfully executed an experiment in which we used growing bacterial and fungal cultures for generating naturally appearing virtual terrains in real-time. Concludingly, we confirm the feasibility of using living organisms in real-time non-behavioral PCG and reflect on its potential impact.

OPEN ACCESS

Edited by:

Raphael Kim,
Queen Mary University of London,
United Kingdom

Reviewed by:

Gustavo A. Patow,
University of Girona, Spain
Carlos Andújar,
Universitat Politècnica de
Catalunya, Spain

*Correspondence:

Maarten H. Lamers
m.h.lamers@liacs.leidenuniv.nl

Specialty section:

This article was submitted to
Human-Media Interaction,
a section of the journal
Frontiers in Computer Science

Received: 30 November 2021

Accepted: 14 February 2022

Published: 08 April 2022

Citation:

Lamers MH and van Eck WJOM
(2022) Microbial Content Generation
for Natural Terrains in Computer
Games.
Front. Comput. Sci. 4:826412.
doi: 10.3389/fcomp.2022.826412

Keywords: bacteria, fungi, virtual terrain, games, procedural content generation (PCG)

INTRODUCTION

Generally, in modern computer-based games, terrains are manually designed, possibly with the aid of procedural content generation. Alternatively, they can be algorithmically generated at the start of the game (Hendrikx et al., 2013). Terrains which autonomously evolve while a game is being played are rarely seen. In an earlier study (Lamers and van Eck, 2012) we discussed potential implications of using (non-human) biological organisms within digital games for the game player, game designer, or integrated non-human organism. We posed the possibility of generating an evolving game terrain in real-time, based on the growth of a live biological system, such as a single organism, a collection of (micro)organisms, or even full ecosystems.

Applying *biologically-inspired* algorithms in the development and design of computer games is common. Bio-inspired swarming behavior of digital characters is a commonly applied example in games (von Mammen and Jacob, 2009). Also, Perlin noise (Perlin, 1985), developed to generate fractal patterns as found in nature, is heavily used in the generation of game terrains.

In prior studies, the behavior of non-playable characters was not only based on *simulations* of real organisms. Living non-human organisms *themselves* have been successfully integrated in computer gaming systems for various purposes. Behavior of non-playable characters was derived in real-time from that of non-human organisms such as trees (Young, 2005), hamsters (Tan et al., 2008), single celled organisms (protozoa) (Wilson, 2006), and crickets (van Eck and Lamers, 2006). Using the mechanics of game-based crowdsourcing, human players have gathered data for experiments on living microorganisms (Riedel-Kruse et al., 2011). Similarly, game-based crowdsourcing setups were used to study the possibility of improving animal welfare through digital means (Tan et al., 2008; Driessen et al., 2014). Even computer games were used in an artistic context to create a symbiotic physical contextualization for plants in real-time (Vermeulen, 2007). Although not necessarily in a gaming context, also the

natural sensing and processing capabilities of living organisms were used as part of larger digital systems in various ways (Hertz, 2004; Rinaldo, 2004; van Eck and Lamers, 2013).

The problem context that we selected for experimental organism-driven in-game content generation is that of virtual terrain generation. Since virtual terrains are a key element of many scenes in games and other computer graphics application, ample prior research and tools exist for terrain generation.

As mentioned, virtual terrains have mostly been manually designed, increasingly with aid of procedural content generation methods. Terrains in computer games and simulations are becoming larger and larger, while users also expect constant improvements in realism and detail. Popular game engines such as Unity and Unreal Engine offer their users basic in-engine tools to interactively shape terrains using brushes to elevate or lower parts of the virtual terrain. Digital sculpting software such as Pixologic ZBrush offers digital artists advanced tools to digitally sculpt their envisioned terrain with full manual control. But while a skilled artist can achieve great results, the work required is too much to create expansive terrains. Specialized terrain generating software such as E-on VUE offers options to procedurally generate terrains with great realism. Such software can also simulate geomorphological effects such as erosion or terracing to recreate realistic geological features. A common workflow now is to create the basic shape of a desired terrain in a digital sculpting program, import this into a program such as VUE to have realistic geological features generated procedurally, and import that generated terrain (in the form of a mesh) back into the game engine. This offers artistic control over the general shape of the terrain, while having detail generated which would have been too labor intensive to add manually.

In cases where digital recreation of real-world terrains is desired, photogrammetry (constructing 3D information from multiple 2D aerial photographs of a terrain), or Lidar scanning can be used. Software such as Agisoft Metashape can be used to import such photographic data, and to generate and export a 3D mesh to be used in game engines. Galin et al. (2019) provided an extensive review of digital terrain creation methods.

Other than for terrains, procedural content generation (PCG) has been applied to fulfill various game-related needs (Shaker et al., 2016), such as for the generation of weaponry [e.g., *Borderlands* (Gearbox Software, 2009), see Game Informer, 2007], terrains [e.g., *The Sentinel* (Firebird, 1986) and *Astroneer* (System Era Softworks, 2019)], flora and fauna [e.g., *No Man's Sky* (Hello Games, 2016)], dialogue (Paradis, 2019), game design (Cook et al., 2017), and much more. Also, external factors were included to drive content generation. An early example of this is *Vib Ribbon* (Sony Computer Entertainment, 1999), a computer game that uses (non-biological) external factors for generating game levels. This platform game, in which the player must avoid obstacles on the rhythm of music, offers the option to insert an audio CD of choice which is then analyzed and used to generate the obstacles in the game.

However, no prior examples exist of using non-human organisms for game content generation, other than for character behavior generation. In *Lumberjacked* (Young, 2005) the movement of leaves in real trees influences the behavior of

tree-avatars in a basic hack “n” slash-type game. Although virtual trees are typically part of game environments (and not characters) in this particular game they act as opponents within a static environment. In *Metazoa Ludens* (Tan et al., 2008) the physical terrain on which a hamster walks deforms according to in-game state changes. Although the game revolves around a living organism and includes a physically changing environment, the environment is not evolved but changes parametrically and not as the direct result of the non-human organism's behavior.

A work that most closely relates to our study is that by Kim et al. (2018). Based on techniques described in the publication of our initial results (van Eck and Lamers, 2015), it created a game for which fungal and bacterial growth was imaged over time. Players online compete in claiming areas depicting microbes in the images. Although the growth patterns of the microbe cultures are used as content within the game, we place this example outside the scope of “content generation,” since the images were used directly as content themselves. This article and our previous work on biological behavior generation for game characters (van Eck and Lamers, 2006), takes the approach of capturing organism behavior and procedurally translating that into virtual content.

To initiate broader study into using non-human organisms for in-game content generation, we experimented with the possibility to apply biologically generated patterns to the generation of in-game terrains, as opposed to possibly simulating natural processes to this end. In fact, we explored the concept of evolving game terrains in real-time through intermediation of living microorganisms. A proof-of-concept realization thereof is presented further in this text.

The method adds an element of unpredictability to the game. It is difficult to predict the development or behavior of a living organism, which we experienced profoundly ourselves when studying crickets controlling characters in a *Pac-Man* style game (van Eck and Lamers, 2006). In fact, the potential values of unpredictability in organisms' behavior or growth, and how they affect in-game mechanics, were identified by Kim et al. (2019). To the context of our study, unpredictably holds potential value as what they term “a precursor for an experience-enhancing game component.” In effect, this unpredictability may transfer to the generated terrains, with unknown and therefore potentially beneficial consequences.

Moreover, the proposed method has the prospect of evolving terrains even when the computer is powered off, which could add a potentially interesting element of real-time behavior to the in-game environment. Community simulation games such as *Harvest Moon* (Nintendo, 2001), in which the game continues even when the player is not actively playing may benefit from this.

Such an approach to in-game content generation through use of micro-organisms could potentially have implications for terrain appearance, for example in terms of naturalness and recognition. But also it could affect player experience, either related to awareness of the biological force behind terrain generation or related to aspects of predictability of terrain development over time. Although understanding such implications is relevant, the current work does not wander into user-experience studies, but aims to investigate the general feasibility of terrain generation using micro-organisms and to

initiate broader study into using non-human organisms for content creation.

It is worth noting that our experimental methods draw inspiration from *DIY science*: an umbrella-term generally covering scientific experimentation outside academic institutes by interested amateurs, often involving unconventional or home-brew equipment and tools. Our experiment was conducted within Leiden University as part of the PhD research of one of the authors. Nonetheless, we drew inspiration from DIY science methods to use tools in unconventional and very cost-efficient manners. This aspect of our experiment plus its implications are discussed in the appropriate sections.

In the remainder of this article, we first look at how game terrains are created using traditional PCG techniques. Then, we describe how fungi and bacteria can be used to generate evolving game terrains and motivate the choice for using microorganisms in the experiment. Image capturing methods are discussed, as are methods of translating images into game terrains. Although our work is highly exploratory from a scientific perspective, we aim for it to broaden the scope of content generation for gaming. Conclusions from our experiment within this broader context are to be found in the discussion section.

PROCEDURAL GENERATION AND EVOLUTION OF GAME TERRAINS

Among the first computer games to use procedural content generation was *Beneath Apple Manor* (Don Worth and The Software Factory, 1978), in which rooms of a dungeon are algorithmically generated each time the game restarts. Its PCG mainly adds replay value; every time the game is played the environment differs slightly. Other games such as *Minecraft* (Mojang, 2009) generate a new natural terrain each time the game restarts. Hendriks et al. (2013) provide a comprehensive survey of PCG for games based on a six-layer taxonomy of game content types (*bits, space, systems, scenarios, design, and derived*). Within their taxonomy, algorithmically generated natural terrain is of content type “Space: Outdoor Maps.”

Unsurprisingly, many different types of terrains occur in computer games. When considering natural terrains, such as mountains, valleys, fields, and such, some games aim at natural realism whereas others appear more abstract. For both types, a common element of natural terrains is height variation. Within 3D animation packages and game engines such height variations in terrain are commonly generated through a *heightmap* or *discrete heightfield*, a matrix in which each cell represents the local height of terrain. Heightmaps are often created using algorithms based on pseudo-random number generation.

Frustrated with the limited realism of procedurally generated natural textures, Perlin (1985) developed what is known as “Perlin noise,” a pseudo-random number (noise) generation algorithm. It enables simulation of organically appearing textures with fractal features, such as landscapes, waves, clouds, even marbled patterns. The noise pattern itself has a pseudo-random appearance, in which visual details are of similar size. By repeatedly stacking noise patterns at decreasing scales

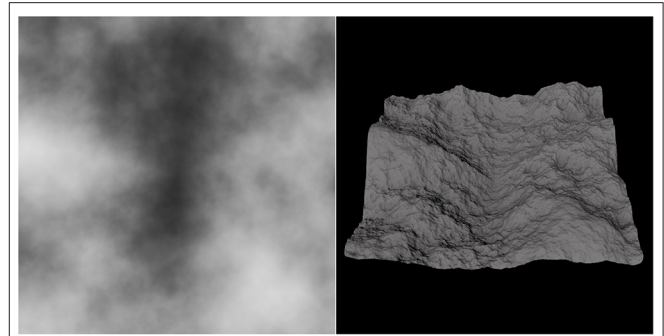


FIGURE 1 | Grayscale image sized 512 × 512 pixels, formed by progressively stacking seven layers of Perlin noise patterns at decreasing scales and intensity (**Left**). Corresponding virtual terrain formed by using the grayscale image as heightmap, where greater luminance leads to higher terrain (**Right**).

and intensities, fractal-like features are formed that resemble naturally appearing phenomena (**Figure 1**). These are commonly used in PCG of natural textures. For example, when using such stacked noise patterns as heightmap, natural appearing virtual terrains are formed, a technique commonly used in games.

A generative method that is commonly applied to form organic and fractal structures is that of L-system grammar term-rewriting, as discussed in-depth by Prusinkiewicz and Hanan (1989). Although mostly applied to the generation of organically growing structures (e.g., plants), it can also be used for generating terrain features (e.g., Ashlock et al., 2005), including underground cave systems (Antoniuk and Rokita, 2016). Given the nature of our proposal, one could consider using L-systems to generate the structural properties of bacterial and fungal cultures, and to subsequently use them as data on which to base terrain generation. Naturally, one might then just as well bypass the concept of microbial growth altogether and generate terrain features directly using L-system grammar, which would contradict the aim of our work.

Broadly, terrain generation methods fall into three categories, as proposed by Galin et al. (2019): (1) procedural or algorithmic generation; (2) techniques that simulate naturally occurring land formation and shaping processes, such as erosion; and (3) example-based methods that rely on scanned terrain data. Techniques may differ along other dimensions also, such as that of generated landform types, geomorphological realism, scale limitations, user control, and rendering speed. A very extensive overview and discussion of terrain generation methods, dissected along multiple dimensions of interest is provided by Galin et al. (2019). Within their taxonomy, our approach applies an elevation model, based on discrete heightfields. One could argue whether it is procedural generation, since it aims at reproducing only the effects of natural physical processes instead of simulating these processes, or whether it takes an example-based approach, in that it borrows from external existing structures, which in our case are microbial cultures and not existing terrains.

Provide designers authoring control over the generated terrains can be a complex matter. Several approaches were developed that provide user control to a large or lesser extent.

Some provide the designer with an interactive workflow that involves efficient generative content creation. Others allow designers different levels of control over the terrain generation, from local control to full automation (De Carpentier and Bidarra, 2009; Doran and Parberry, 2010; Galin et al., 2019). An extensive overview is provided by Hendriks et al. (2013). We do not explicitly aim at providing users with control over the generative process beyond offering indirect control through implicit tuning parameters, such as discrete heightfield scaling.

It is uncommon to include an *evolving* generated terrain, one that continues developing while the game is played. During the development of *Fable* (Big Blue Box, 2004), well-known game designer Peter Molyneux announced that it would feature a constantly evolving world in which you could for example plant an acorn and then watch it grow into an oak tree over the course of the game (Schreier, 2014). Although this feature was indeed coded by the developers, it supposedly took 15% of the total processor time and did not appear in the published game. Moreover, it did not appear in later games of the *Fable* series which would run on more powerful hardware. Although some terrain modeling tools provide users the possibility of time-evolving terrain, by placing events on a timeline, we do not yet consider such user interactions and primarily aim at terrain evolution following the evolution of microbial cultures in real-time. Moreover, according to Galin et al. (2019) such time-line control is currently only available in methods based on simulation of geomorphological processes, and not in current procedural or example-based methods.

GENERATING TERRAINS FROM MICROBIAL CULTURES

We aim to experiment with generating patterns for constructing naturally appearing virtual terrains, through intermediation of living organisms as opposed to using algorithmic approaches. Additionally, we place this aim within the problem context of evolving game terrains in real-time, while the organisms are actually growing. A proof-of-concept realization was developed and described in this section.

Our experimental method is based on the before-mentioned concept of using heightmaps to describe height variations in terrain. The heightmaps are generated from images of growing organisms. We hypothesize that the organisms' visual growth exhibits patterns such as fractals, symmetries and spirals that make height variations in terrains appear natural. This informal hypothesis is motivated when discussing what organisms are used, further on. Although it is technically possible, we discarded the idea of using sound or bioelectrical activity produced by organisms to generate height maps, since they yield one-dimensional information which is less suitable for generating height variations in virtual terrains. Moreover, these signals are beyond our own scope of feasibility for this experiment.

Basic components required for our experiment are growing (micro)organisms, a method for visually capturing that growth over time, plus a method for translating the captured images into virtual terrains. Drawing inspiration from DIY science,

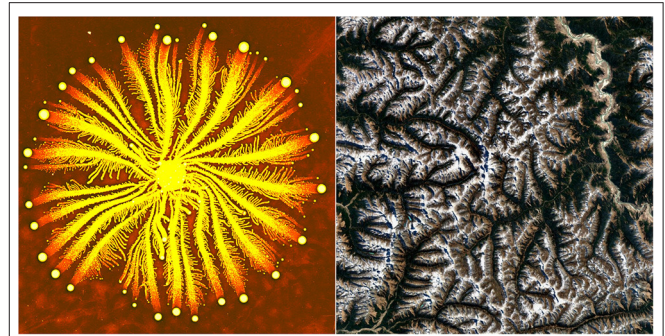


FIGURE 2 | Illustration of morphological similarity between a microbial culture and natural landscape. A colony of *Paenibacillus vortex* bacteria (image by Prof. Eshel Ben-Jacob, CC BY-SA 3.0) (Left). Mountainous landscape in Mainling, Nyingchi, Tibet, as seen on Google Earth (Right).

we were explicitly open to accepting unconventional methods when they provide cost-efficiency or other benefits over more accepted tools and methods. Here we discuss and motivate the methodological choices made for each of these components, plus their potential impact.

Growing Bacteria and Fungi

When choosing a living organism for the experiment, several factors are important to our intended use: availability, growth speed, growth patterns, ease of cultivation, ease of imaging, and ethical considerations. Furthermore, the authors of this study are not biologists, excluding organisms that require laboratory-type conditions and handling. Vertebrate animals, or more generally locomotive animals, were not considered for use in our experiment for practical reasons. Parts of plants, such as pinecones, seed heads of flowers, *Romanesco broccoli* and tree branches display interesting growth patterns, including spirals, symmetries, and fractals that may be found in natural landscapes also (and that are commonly replicated using L-System grammars). However, imaging them may be difficult since they cannot be easily contained on a fixed spot while growing, a requirement for repeated imaging over time. Moreover, the slowness of their growth would be impractical for our experiment, although it does not fundamentally exclude them from future use.

Cultures of bacteria and fungi are easily grown on growth medium contained inside a Petri dish. They can display aforementioned growth patterns, while colonies of bacteria such as *Paenibacillus vortex* can resemble the structures of mountain ranges as seen on satellite imagery in a remarkable way (Figure 2). Moreover, bacteria and fungi are very easy to obtain: Petri dishes and growth medium prepared in a non-sterile environment are likely contaminated with various microbes.

We grew cultures inside Petri dishes on a self-made growth medium that provided a nutritious base for our microbes. It was made of agar, to which we added honey, Marmite, powdered milk, and activated carbon powder. After some prior experimentation with ingredients that included instant coffee powder, activated carbon powder was chosen for coloring the growth medium

black, providing good visual contrast with the microbial cultures. Petri dishes were filled with growth medium up to 2 mm from the top of the dish, for reasons that are explained further on.

The growth medium and Petri dishes were prepared and kept in a non-sterile and non-laboratory environment. No microbes were purposefully introduced on our growth medium, but contamination is introduced *via* the ingredients, preparation, handling and exposure to the environment. In our experience the first visible cultures appeared after roughly 2 days, followed by steady growth of roughly 6 days duration, after which the cultures generally stabilized. These timeframes can deviate depending on ambient temperatures and types of bacteria and fungi. Higher and lower temperatures can speed up or slow culture growth, while 37°C is generally regarded as an optimal growth temperature.

Given the lacking background of the authors in microbiology, we cannot identify specific bacterial and fungal cultures appearing in the dish. However, identification of species is not essential for our purpose, since non-deterministic variation in terrain output is inherent of the proposed approach. When desired, for example to ensure the presence of salient terrain features, specific bacteria and fungi can be cultivated by transferring them from one Petri dish to another using a swab. Moreover, desired terrain structures at specific locations could in principle be attained by introduction of specific microorganisms at exact locations within the Petri dish.

Visual Capturing of Growing Cultures

To visually capture the growth of our bacterial and fungal cultures consistently over time, our imaging method must be stable with regard to culture positioning and lighting. In practical terms, this implies it requires consistent lighting, fixed exposure, fixed white balance, minimal imaging noise, and the setup should not move in-between captures. Any deviation from these requirements introduces visual artifacts that are not part of the imaged cultures themselves, and as such they are unwanted as they would affect the generated terrain. Given the scale at which structures appear in the individual cultures, high-resolution imaging is required.

In the aforementioned spirit of DIY science, the imaging device used was a consumer flatbed document scanner. Such devices are priced well below microscopes and fancy digital cameras (and as we noticed, they are easily found in thrift stores). To visually capture what lies on its horizontal glass plate, an internal light source and sensor move progressively below and across the plate while the sensor captures what lies on the glass plate. The internal light source satisfies our requirement for consistent lighting. Moreover, since the device moves its sensor to capture light intensities locally at each position on the plate, all pixels in the resulting two-dimensional image were captured with the sensor orthogonal to the plate. This makes the resulting images represent objects at varying distances from the glass plate in a consistent manner. Sensors that use CCD (charged-coupled device) technology offer a greater depth of field than those that employ CIS (contact image sensor) technology. As a result, scanners with CCD sensors maintain better focus on objects that are not in contact with the glass plate, and therefore

capture such objects more accurately. For our experiments, we used the Epson Perfection V370 Photo flatbed scanner, which uses CCD sensor technology. We manually underexpose the scanner slightly to prevent artifacts caused by the sensor's overexposure.

Somewhat to our surprise, document flatbed scanners were not commonly used for scientific imaging. Although they may be obviously used to capture images of specimens for storing, sharing, and archiving, we could find only few examples of their use as part of experimental methodology. Mims and Mims (2004) used them to scan colony forming units of bacteria and fungal spores in airborne smoke captured on 3M Petrifilm (ready-to-use agar film), to enable easier visual access for (human) counting and identification of colony forming units. Colony growth was not captured over time. Using a modified document flatbed scanner Hossain et al. (2015) created time-lapse videos of Petri dishes with slime mold *Physarum polycephalum*'s chemotactic reaction to liquid oatmeal solution dispensing. Emerging from the DIY science community was a proposal by Damase et al. (2015) for converting an LED-based document flatbed scanner to enable fluorescence detection in DNA nanotechnology research. In similar spirit, Gaudenz (2013) hacked a flatbed scanner to analyse gels of stained DNA and proteins into a 1980's Delsey business briefcase.

When our Petri dishes are ready to be scanned, we remove their lid and place them upside-down on the glass plate of our scanner (Figure 3). When the Petri dishes are filled with growth medium up to 2 mm from the top, and when placed upside-down, the cultures are near enough the glass plate to be captured in focus. The 210 x 297 mm sized glass plate of our scanner fits six Petri dishes, which can be simultaneously scanned at a resolution of 1,200 dpi with full color depth. This results in scans of roughly 9,000 × 13,000 pixels. A single Petri dish image cropped from the full scan is sized 4,200 × 4,200 pixels. Higher resolution scans are possible with our scanner, but were not required for our purpose. A single scan takes 3 min to complete. For capturing the growth of organisms over time, we periodically scan the six Petri dishes at intervals of 5 min.

To gather data for our experiment, we started scanning 1 day after the growth medium was poured into the Petri dishes, allowing the agar to cool to ambient temperature. Periodic scanning was terminated when the growth of the cultures appeared to have stagnated, or could have been terminated when the growth medium dried out. The total duration of our data gathering was 7 days and 22 h, resulting in 2,282 captured images.

The resulting images are of sufficient resolution, free of artifacts, and evenly illuminated (Figure 4). Unfortunately, the light emitted from the scan head is reflected by glossy surfaces, resulting in unwanted bright artifacts appearing in images on glossy bacteria cultures. While not all bacteria cultures have glossy surfaces it would be preferable to work only with (less glossy) fungi. We noticed that growth medium made of agar, honey, and instant coffee powder reduced the number of bacterial cultures, although a sterile setup would be required to entirely avoid them. We chose to continue working with both fungi and bacteria, retaining our simple and affordable *DIY science* setup.

Moreover, the bacteria cultures exhibit growth patterns that we did not want to exclude from our experimentation.

Images Sequences and Heightmaps

Potentially numerous ways exist to generate virtual terrain from our captured images. This study does not aim to find the best technical method to accomplish this. What is *best*, is at least partially determined by requirements from the intended use for the terrains. By exploring a single method, we present a proof-of-concept for the general approach of generating natural terrains from growing bacterial and fungal cultures. For the same reason also, the generated terrains need not necessarily be visualized within a game engine in real-time, but can be generated after the visual capturing was completed. Although we did create a real-time setup using the Unity game engine, we used 3D modeling and animation package Maxon Cinema 4D to render still images and movies, making it easier to document our results.



FIGURE 3 | The Epson Perfection V370 Photo flatbed scanner used in our experiment, with several Petri dishes upside down on the scanning area. One dish is turned upright to show considerable microbial growth. Different growth medium colors are the result of experimentation with ingredients that included instant coffee powder or activated carbon powder.

When reviewing the captured images of our experimental dataset, we decide to update the visualized terrain only once every capture-hour, which translated into using one in every 12 captured images. This decision was based on practical issues involving the handling of quite significant data volumes, plus the observation that differences between consecutive captures are very small from a visual perspective. This resulted in 191 captures, visualizing the total scanning period of 7 days and 22 h.

In line with common game creation workflows for terrain visualization, we derive heightmaps from our captures by converting them into 8-bit (256 level) grayscale values. Since the brightly colored cultures contrasted well with the black growth medium in the Petri dish, the captures were well-suited for such use. A heightmap resolution of 512×512 pixels is common among popular game engines and sufficient for our proof-of-concept experiment. The visual capture of a single Petri dish measures $4,200 \times 4,200$ pixels, so we could either downscale a section of a captured image, or use a 512×512 -pixel crop. Working with pre-captured images offered the advantage of knowing in which section of the Petri dish cultures formed and how they are shaped. In a real-time setup, one could contaminate a specific area of the Petri dish to ensure that a certain culture will grow on that area.

We selected an area within the captured images that featured both fungi and glossy bacteria, from which 512×512 -pixel images were cropped. The cropped images were subsequently converted to 8-bit grayscale representation. The resulting stack of images was exported into a sequence of 191 PNG-format image files.

Generating Terrain

By importing the sequence of images files as an animated heightmap into the displacement channel of Maxon Cinema 4D, a three-dimensional visual representation of a terrain was created in which greater luminance within the image resulted in greater height of the terrain at the corresponding spatiotemporal location (**Figure 5**).

The direct mapping of grayscale values to terrain height resulted in sudden terrain peaks at spatial locations where



FIGURE 4 | One $13,000 \times 9,000$ -pixel full capture of our scanner that shows microbial cultures in six Petri dishes (**Left**). The top-right Petri dish cropped from the full capture, resulting in a $4,200 \times 4,200$ -pixel image of the single dish (**Middle**). Cropped 512×512 -pixel section of the selected Petri dish showing further structural detail of the growing cultures (**Right**).

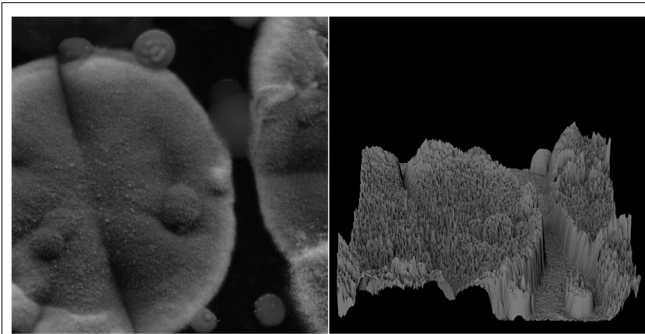


FIGURE 5 | Grayscale 512 × 512-pixel section of a microbial scan (Left). Shape of corresponding terrain when directly mapping grayscale values to height, including sudden peaks at spatial locations with high luminance (Right).

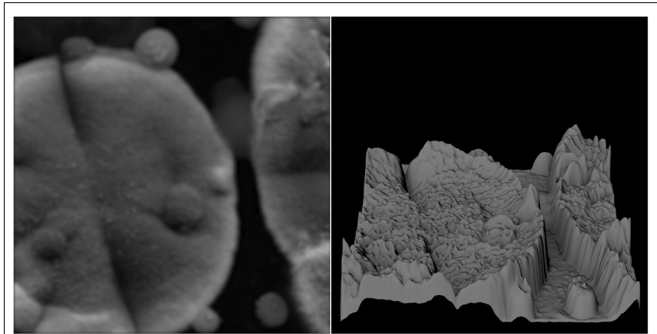


FIGURE 6 | Grayscale 512 × 512-pixel section of microbial scan after light blurring (original shown in Figure 5) with a low-width blurring filter (Left). Shape of the corresponding terrain, showing strongly reduced terrain noise (Right).

glossy matter resulted in specular reflection of the light emitted by the document scanner. Also, it is notable that over time the terrain height did not increase monotonously, but both rising and lowering of terrain height occurred. Moreover, the resulting terrain undulated strongly with high spatial frequency, a phenomenon referred to as “terrain noise” in broader terrain-developing communities. This makes the resulting terrain not easily accessible for ground-traveling characters within a game. Reducing the level of vertical terrain displacement in response to variations in heightmap values is simple, and results in terrain that varies less in height. Nonetheless, relatively high terrain noise gives the terrain an unnatural appearance.

Although such unnatural appearance may be used for deviating alien landscapes, underwater terrains, or environments free of atmospheric erosion such as planetary landscapes, generally a more accessible and earthlier appearing terrain is desired. To this end, one could apply some form of processing on the heightmap. One method is multiscale stacking of heightmaps, a technique commonly applied using Perlin noise to create heightmaps and discussed in section Procedural Generation and Evolution of Game Terrains. A simpler approach would be to smoothen the heightmaps (which are effectively grayscale images) using a low-width blurring filter (Figure 6). The latter approach results in reduced terrain noise and heightmaps corresponding structurally more closely to the original, and therefore to the microbial cultures’ structure. Multiscale stacking leads to increased structural changes. It should be noted that large peaks and troughs caused by unwanted artifacts in the image capturing process cannot be fully eliminated by smoothing. Interestingly, smoothing holds similarities to erosion of natural landscape terrain.

Terrain “naturalism” is a somewhat undefinable quality. Nonetheless, Rajasekaran et al. (2019) made a first step in assessing perceptual realism of generated terrain models. Through a user-study that compared perceived realism of real and generated terrains. From the results, they were able to gather quantitative insight into what local morphological features affect perceived realism. Although the method is highly promising, it does not apply well to larger scale structures such as large valleys

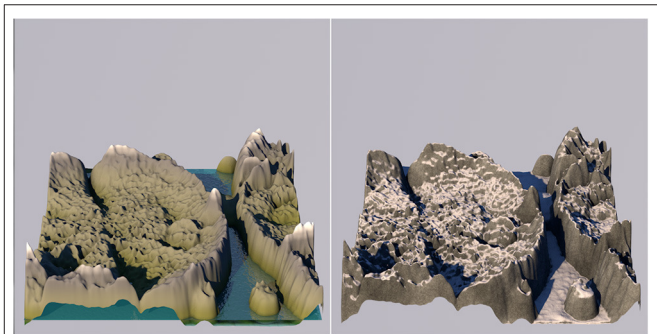


FIGURE 7 | Smoothed terrain as shown in Figure 6, with procedurally added natural elements using very basic operations. Different colors were assigned to terrain elements based on local terrain height, and water was added (Left). Slope angle information was used to accumulate virtual snow (Right).

and depends on the scale at which a terrain is viewed. To the best of our knowledge this method was not yet applied within other studies, nor is an implementation yet available for application.

Let’s consider the related concepts of recognizability. Recognizability can be increased by adding elements that are commonly found in natural terrains, such as snow, vegetation, and water. When considering procedural methods, one simple approach is to apply common shaders. For example, slope-angle information could be used to accumulate virtual snow on peaks and horizontal planes. Or different terrain coverings textures (e.g., water, grass, rock, snow) can be assigned to predefined height ranges (Figure 7). In many real natural terrains, water bodies are commonly found below certain terrain heights also.

Furthermore, the color information (besides only luminance data) from the captured images could be used. Several possible ways of using the images’ color information could be considered. Firstly, the generated terrain could be colored using the color at the spatiotemporal image position from which its height is derived. In that case, the terrain would reflect the microbial cultures’ changing colors over time and space (Figures 8, 9, videos available on publisher’s website).

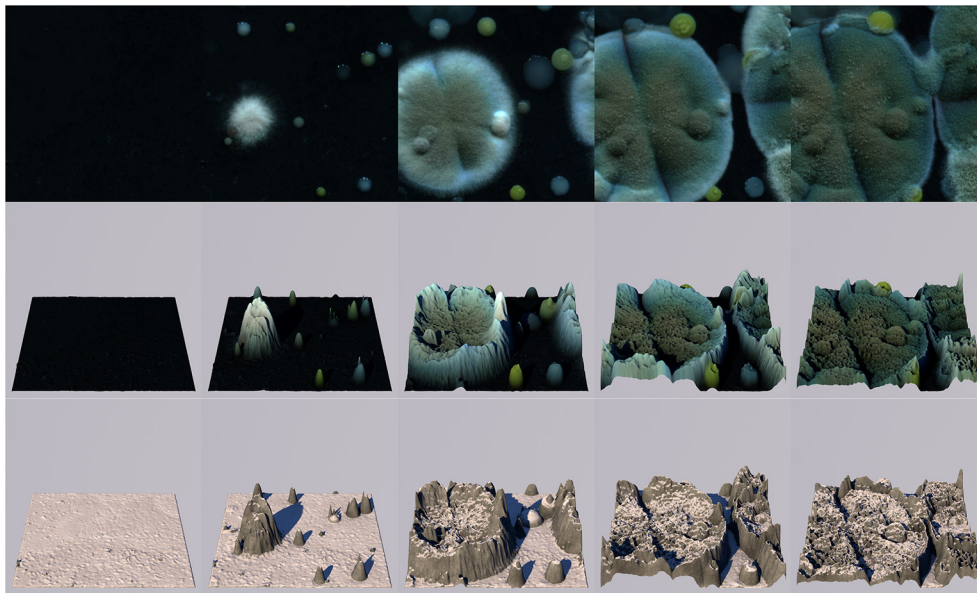


FIGURE 8 | Five stages of the growth process with roughly 2-day intervals, from left to right. The top row shows the microbial scans (same region as in **Figures 5–7**); the middle row shows the generated terrain colored by the images' color data; the bottom row shows the terrain with accumulated virtual snow.

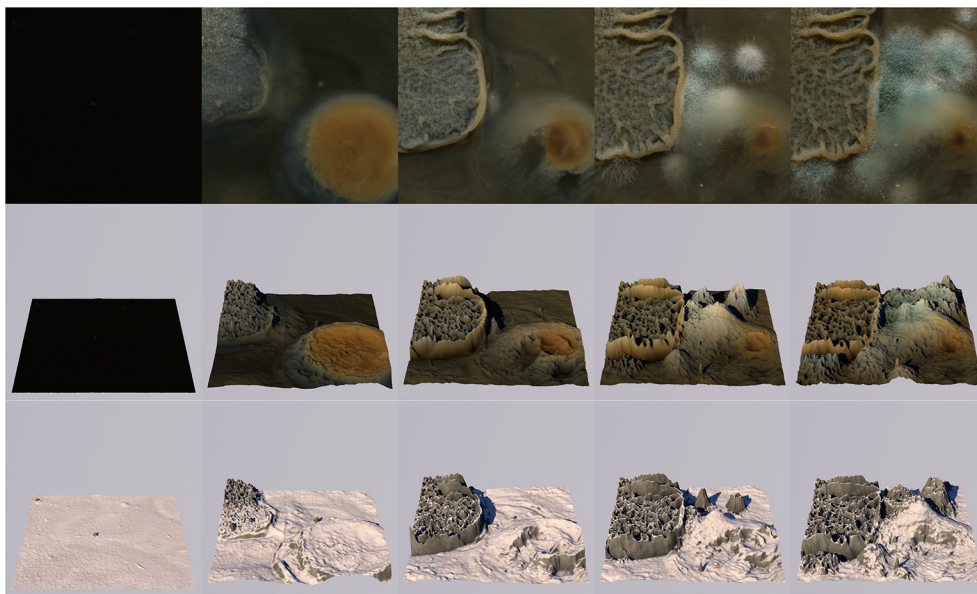


FIGURE 9 | Another example of five stages of the growth process with roughly 2-day intervals. Columns (temporal stages) and rows (scans and generated terrains) depict the same parameters as in **Figure 8**, but for another microbial scan.

This could be considered in situations where player awareness of the relation between the terrain and microbial growth is desired. Educational aims and player experience could motivate this choice. Alternatively, color information from captures could be used to derive other local features of the terrain, such as soil type, rock type, vegetation type, or to assign game-specific location-based features such as portals or special features.

Natural realism of terrain shapes could be increased while maintaining global terrain structure as much as possible. For example, by procedurally adding to the terrain multiple Perlin noise patterns at relatively small and decreasing scales and intensities. That way, smaller scale fractal-like features are added to the overall height variations that resemble naturally appearing terrain textures, while larger scale microbial structures remain prominent.

DISCUSSION

As argued in the introduction, procedural content generation is commonly applied for the design of in-game content, including (natural) terrains. Examples exist of PCG for games that use external factors on which they base content generation. Moreover, living (non-human) organisms were used for real-time behavior generation in games. However, no prior work exists in which living organisms were used as part of procedures for generating non-behavior content, let alone in real-time.

We successfully executed a proof-of-concept experiment, in which we used growing bacterial and fungal cultures for generating naturally appearing virtual terrains. Moreover, we chose a problem context of evolving terrain for games in real-time, while the organisms are actually growing. With our work, we confirm the feasibility of using living organisms in real-time non-behavior PCG.

An aspect to review is the natural appearance of the generated terrains. To our own geologically untrained eye, they appear as possible results of geological processes. A users' experience study could be undertaken to properly assess this quality, preferably combined with methods for assessing perceived terrain realism as proposed by Rajasekaran et al. (2019). Such a study could distinguish between general appearance and recognizability as natural terrain, and would require elaborate texturing and terrain type assignment when comparing the outcomes to commercially used outcomes or existing natural terrain. This falls, however, outside the scope of our proof-of-concept experiment, that was unavoidably undertaken in the form of a case study, whereby one specific case was investigated. We trust that our description of the work allows the method to be re-applied in future user-studies.

Similarly, how users experience the natural growth over time of microorganisms in the generated terrain, possibly in relation to its real-time nature, could be investigated in a user-study. Parameters that could be scrutinized in such a study are the reflection of organisms' properties within the terrain (such as its color as discussed in section Generating Terrain), player locomotion types, added natural elements, and more.

Focussing more on the player themselves, a follow-up study could investigate whether player experience is different when exploring a terrain with prior knowledge that it is derived in real-time from a living organism, compared to knowing that it was algorithmically generated, or even a duplicate of real-world terrain. With respect to in-game opponent characters, it was shown by Weibel et al. (2008) that players who falsely believed they played online games against human-controlled opponents experienced more enjoyment, presence and flow in doing so, as opposed to subjects aware of the opponent's true non-biological (algorithmic) control. A similar "awareness-effect" could be hypothesized to exist for (real-time) biologically grown game terrains.

In instances of real-time microbial terrain generation, interaction of the virtual game state with the real microbial cultures is possible. Player behavior or gameplay mechanics could stimulate, delay or stop growth at certain locations in the Petri dish through changes in the exposure to light, temperature, nutrition or toxics (e.g., alcohol or antibiotics). Such influencing

of the growing organisms enables also reverse causation, where growth patterns affect game states directly potentially acting as opponent or co-player.

As with every case study or proof-of-concept, one must place the findings in a context of broader significance. This holds for both the specific experiment and generalizations of its method. Clearly, practical applicability has to this point not been a concern of the authors. It may be hard to imagine a future context in which content for games is generated from living organisms for reasons of efficiency, quality, or cost. For reasons of practicality, player-side microbial content generation may be even harder to imagine, where the microorganisms reside within the player's local gaming system.

Nonetheless, it would be possible with relatively little effort to setup a central "microorganism grow house" server from which captured images are streamed in real time. These images could be streamed both in their raw form and simultaneously in pre-processed formats. The output may then be used locally by client systems for terrain generation. Alternative client applications could use the server output for other purposes, such as generative art and design, education, even research. It is noteworthy that in our experimental setup, a single Petri dish can accommodate 37 unique commonly sized 512×512 -pixel heightmaps simultaneously. Given that in our simple setup six Petri dishes develop simultaneously on a single document scanner, our limited approach could already generate and stream 222 unique 512×512 -pixel terrain heightmaps to different clients simultaneously in real time. This would offer an alternative to player-side growing and capturing of microorganisms.

A realm in which the proposed method could have significance beyond our single experiment is that of education. Observation of real time growing behavior of microorganisms can contribute to their understanding. Identification of colonies, growth patterns, effects of external factors (temperature, growth medium composition, humidity, etc.) are concepts that could find useful application in education, be it within or without translation into a gaming context. Kim et al. (2020) conclude from organizing two successful workshops on ideation of bio-hybrid games that working with games involving microbes can enhance the teaching and learning experience for areas as biotechnology.

Our experiment has also led to several by-products, or unintended secondary results. Since it was not a prior-specified aim of our study, the method of visually capturing growth of microbial cultures can be considered a by-product. Use of a consumer document flatbed scanner for capturing growth or organisms over time was to the best of our knowledge novel at the time of our experiment in Summer 2014. In combination with our black growth medium, this provides a cost-friendly method to obtain very high-quality images of six Petri dishes containing microbial cultures simultaneously. Cost-efficiency and ease of the method could enable it to enter DIY science practices, and possibly stimulate its use in (informal) education and *maker* communities. In institutional science, our method has since its initial publication been successfully applied to capture microbial grow by Kim et al. (2018).

Another by-product of the experiment was media attention; not for the experiment as such, but for the resulting time lapse recordings of growing microbial cultures. After making public several time lapse videos on YouTube.com, we were contacted by media production companies and cultural institutions requesting to use our videos in various public forms. As a result, they have appeared as part of Danish event “Zetland Live #12” (October 2017) on the topic of pandemics. But also in theater production “In Dir Schläft ein Tier” (November 2017) of the Junges Staatstheater Berlin, a documentary named “The History of Food” (November 2018) which aired on the online streaming service Curiosity Stream, various music videos of independent artists, and as illustrations on various websites.

With our work we trust to have forwarded the study of integrating non-human organisms in digital contexts. The problem context that we chose for our experiment is that of procedural content generation for digital applications such as games. We have confirmed the feasibility of using growing microbial cultures in real-time generation of virtual terrain.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

REFERENCES

- Antoniuk, I., and Rokita, P. (2016). “Generation of complex underground systems for application in computer games with schematic maps and L-systems,” in *Computer Vision and Graphics, Lecture Notes in Computer Science 9972*, eds L. Chmielewski, A. Datta, R. Kozera, and K. Wojciechowski (Cham: Springer), 3–16. doi: 10.1007/978-3-319-46418-3_1
- Ashlock, D. A., Gent, S. P., and Bryden, K. M. (2005). “Evolution of L-systems for compact virtual landscape generation,” in *Proceedings of the IEEE Congress on Evolutionary Computation*, Vol. 3 (Edinburgh: IEEE), 2760–2767. doi: 10.1109/CEC.2005.1555041
- Cook, M., Colton, S., and Gow, J. (2017). The Angelina Videogame design system: part I. *IEEE Trans. Comput. Intell. AI Games* 9, 192–203. doi: 10.1109/TCIAIG.2016.2520256
- Damase, T. R., Stephens, D., Spencer, A., and Allen, P. B. (2015). Open source and DIY hardware for DNA nanotechnology labs. *J. Biol. Methods* 2, e24. doi: 10.14440/jbm.2015.72
- De Carpentier, G. J. P., and Bidarra, R. (2009). “Interactive GPU-based procedural heightfield brushes,” in *Proceedings of the 4th International Conference on Foundations of Digital Games* (Orlando, FL), 55–62. doi: 10.1145/1536513.1536532
- Doran, J., and Parberry, I. (2010). Controlled procedural terrain generation using software agents. *IEEE Trans. Comput. Intell. AI Games* 2, 111–119. doi: 10.1109/TCIAIG.2010.2049020
- Driessen, C., Alfrink, K., Copier, M., Lagerweij, H., and van Peer, I. (2014). What could playing with pigs do to us? *Antennae: J. Nat. Visual Cult.* 30, 79–102.
- Galín, E., Guérin, E., Peytavie, A., Cordonnier, G., Cani, M.-P., Benes, B., et al. (2019). A review of digital terrain modeling. *Comput. Graph. Forum* 38, 553–577. doi: 10.1111/cgf.13657
- Game Informer (2007). *Borderlands, The Final Frontier*, Game Informer, Vol. 173. Sunrise Publications, 64–73.
- Gaudenz, U. (2013). *DIY Mobile Gen Lab Transilluminator*. Available online at: <http://www.gaudi.ch/GaudiLabs/wp-content/uploads/DIYMobileGenLab.pdf> (accessed November 29, 2021).

AUTHOR CONTRIBUTIONS

ML contributed to conceiving of the study, steering of the experimentation, and writing of the report. WE contributed to conceiving of the study, execution of the experimentation, and writing of the report. All authors contributed to the article and approved the submitted version.

ACKNOWLEDGMENTS

We are thankful to Peter van der Putten and Roland van Dierendonck for their suggestions when writing this article. A concise description of our experiment was initially presented and published at the 4th International Conference on Evolutionary and Biologically Inspired Music, Sound, Art, and Design (van Eck and Lamers, 2015).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fcomp.2022.826412/full#supplementary-material>

- Hendrikx, M., Meijer, S., van der Velden, J., and Iosup, A. (2013). Procedural content generation for games: a survey. *ACM Trans. Multimedia Comput. Commun. Appl.* 9, 1. doi: 10.1145/2422956.2422957
- Hertz, G. (2004). *Cockroach Controlled Mobile Robot*. Available online at: <http://www.conceptlab.com/roachbot/> (accessed November 29, 2021).
- Hossain, Z., Jin, X., Bumbacher, E. W., Chung, A. M., Koo, S., Shapiro, J. D., et al. (2015). “Interactive cloud experimentation for biology: an online education case study,” in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul: ACM), 3681–3690. doi: 10.1145/2702123.2702354
- Kim, R., Thomas, S., van Dierendonck, R., and Poslad, S. (2018). “A new mould rush: designing for a slow bio-digital game driven by living micro-organisms,” in *Proceedings of the 13th International Conference on the Foundations of Digital Games*, Vol. 10 (Malmö), 1–9. doi: 10.1145/3235765.3235798
- Kim, R., Thomas, S., van Dierendonck, R., Wood, C., and Poslad, S. (2020). Toward growable computer games: insights from biotic game ideation workshops. *ACM Interact.* 27, 82–85. doi: 10.1145/3378563
- Kim, R., van Dierendonck, R., and Poslad, S. (2019). “Moldy ghosts and yeasty invasions: glitches in hybrid bio-digital games,” in *CHI Conference on Human Factors in Computing Systems* (Glasgow), 1–6. doi: 10.1145/3290607.3312895
- Lamers, M. H., and van Eck, W. (2012). “Why simulate? hybrid biological-digital games,” in *Applications of Evolutionary Computation, Lecture Notes in Computer Science 7248*, eds P. A. Castillo, J. Luis, J. Laredo, F. F. de Vega (Berlin: Springer), 214–223. doi: 10.1007/978-3-642-29178-4_22
- Mims, S. A., and Mims, F. M. I. I. I. (2004). Fungal spores are transported long distances in smoke from biomass fires. *Atmosph. Environ.* 38, 651–655. doi: 10.1016/j.atmosenv.2003.10.043
- Paradis, F. (2019). “Procedural generation of cinematic dialogues in assassin’s creed Odyssey,” in *Presentation at Game Developers Conference (GDC’19)* (San Francisco, CA).
- Perlin, K. (1985). An image synthesizer. *ACM SIGGRAPH Comput. Graph.* 19, 287–296. doi: 10.1145/325165.325247
- Prusinkiewicz, P., and Hanan, J. (1989). *Lindenmayer Systems, Fractals, and Plants, Lecture Notes in Biomathematics* 79. New York, NY: Springer. doi: 10.1007/978-1-4757-1428-9

- Rajasekaran, S. D., Kang, H., Benes, B., Cadik, M., Galin, E., Guérin, E., et al. (2019). PTRM: perceived terrain realism metrics. *arXiv [Preprint]*. arXiv:1909.04610v1. Available online at: <https://arxiv.org/pdf/1909.04610.pdf> (accessed September 10, 2019).
- Riedel-Kruse, I. H., Chung, A. M., Dura, B., Hamilton, A. L., and Lee, B. C. (2011). Design, engineering and utility of biotic games. *Lab Chip* 11, 14–22. doi: 10.1039/C0LC00399A
- Rinaldo, K. (2004). *Augmented Fish Reality*. Available online at: <https://www.kenrinaldo.com/portfolio/augmented-fish-reality/> (accessed November 29, 2021).
- Schreier, J. (2014, November 3). The man who promised too much. *Kotaku*. Available online at: <https://kotaku.com/the-man-who-promised-too-much-1537352493> (accessed November 29, 2021).
- Shaker, N., Togelius, J., and Nelson, M. J. (2016). *Procedural Content Generation in Games*. Cham: Springer. doi: 10.1007/978-3-319-42716-4
- Tan, R. T. K. C., Cheok, A. D., Peiris, R., Todorovic, V., Hui Cong Loi, H. C., Loh, C. W., et al. (2008). Metazoa ludens: mixed reality interactions and play for small pets and humans. *Leonardo* 41, 308–309. doi: 10.1162/leon.2008.41.3.308
- van Eck, W., and Lamers, M. H. (2006). “Animal controlled computer games: playing pac-man against real crickets,” in *Entertainment Computing, Lecture Notes in Computer Science 4161*, eds R. Harper, M. Rauterberg, and M. Combetto (Berlin; Heidelberg: Springer), 31–36. doi: 10.1007/11872_320_4
- van Eck, W., and Lamers, M. H. (2013). Hybrid biological-digital systems in artistic and entertainment computing. *Leonardo* 46, 151–158. doi: 10.1162/LEON_a_00530
- van Eck, W., and Lamers, M. H. (2015). “Biological content generation: evolving game terrains through living organisms,” in *Evolutionary and Biologically Inspired Music, Sound, Art and Design, Lecture Notes in Computer Science 9027*, eds C. Johnson, A. Carballal, and J. Correia (Cham: Springer), 224–235. doi: 10.1007/978-3-319-16498-4_20
- Vermeulen, A. (2007). *Biomodd*. Available online at: <http://www.biomodd.net/> (accessed November 29, 2021).
- von Mammen, S., and Jacob, C. (2009). “Swarming for games: immersion in complex systems,” in *Applications of Evolutionary Computing, Lecture Notes in Computer Science 5484*, eds M. Giacobini, A. Brabazon, S. Cagnoni, G. A. Di Caro, A. Ekárt, A. I. Esparcia-Alcázar, M. Farooq, A. Fink, and P. Machado (Berlin; Heidelberg: Springer), 293–302. doi: 10.1007/978-3-642-01129-0_33
- Weibel, D., Wissmath, B., Habegger, S., Steiner, Y., and Groner, R. (2008). Playing online games against computer- vs. human-controlled opponents: effects on presence, flow, and enjoyment. *Comput. Human Behav.* 24, 2274–2291. doi: 10.1016/j.chb.2007.11.002
- Wilson, S. (2006). “IntroSpection and protozoa games,” in *13th International Symposium on Electronic Art*. Available online at: <https://isea-archives.siggraph.org/presentation/introspection-and-protozoa-games/> (accessed November 29, 2021).
- Young, D. (2005). *Lumberjacked*. Available online at: <https://classic.rhizome.org/artbase/artwork/35526/> (accessed November 29, 2021).

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Lamers and van Eck. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.