3 Working groups

3.1 Language Models for Procedural Content Generation

Maren Awiszus (Leibniz Universität Hannover, DE), Alexander Dockhorn (Leibniz Universität Hannover, DE), Amy K. Hoover (New Jersey Institute of Technology, US), Antonios Liapis (University of Malta – Msida, MT), Simon M. Lucas (Queen Mary University of London, GB), Mirjam Palosaari Eladhari (Stockholm University, SE), Jacob Schrum (Southwestern University – Georgetown, US), and Vanessa Volz (modl.ai – Copenhagen, DK)

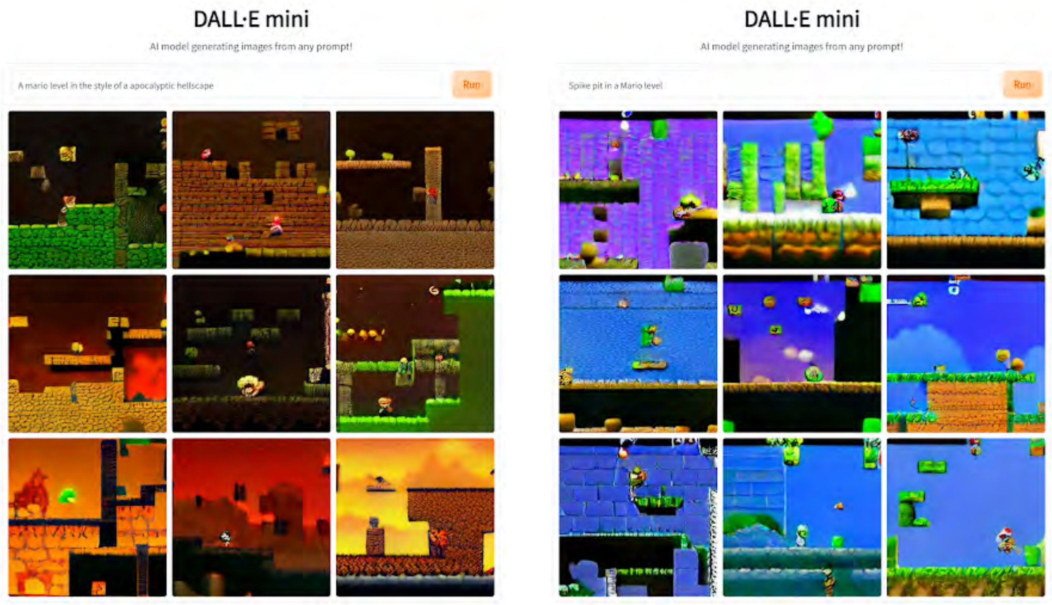
3.1.1 Introduction and Motivation

Recent advances in ML-based image generation via systems like DALL-E [6] beg the question of whether similar tools can be used for generating game content. Specifically, it is desirable to generate game content based on simple text input. As our group was composed of researchers most familiar with level generation, we focused on video game level generation for 2D games first. However, other assets like textures also seem like great examples to use these generative methods on. Our general intuition was, that methods like DALL-E are able to generate impressive previously unseen images due to the strength of the diverse language processing learned with huge datasets of images and their descriptions. For games, in which only little data can be given for a domain like e.g. 2D Super Mario levels, such a large network can not easily be trained from scratch. Therefore, we wanted to investigate the capabilities of a pretrained DALL-E to generate content without any game specific training.

3.1.2 Exploration of Applications

We ran tests using **DALL-E mini** [3] as an intermediate tool for generating content. Figure 1 shows some of those generated examples. Although DALL-E mini can create outputs that look like Mario levels given a prompt like “Mario level”, it has problems incorporating specific details suggested from prompts such as “spikes” or “pipes” in “Mario level with a spike pit” or “Mario level with pipes”. The problem seems to be that DALL-E’s concept of what spikes or pipes are is based on typical photo examples of these items rather than examples of these items in the context of a Mario game. However, prompts that only change the style of the level, like “apocalyptic”, can influence the output. We also did some small preliminary tests on content other than platformer levels, like generating images of new “Pokémon” from a textual description. These examples suffer from similar problems. DALL-E can generate a “Pokémon”, but adding additional descriptors is less likely to be successful. It will be interesting for future work to find out what kind of prompts can and cannot be mixed with DALL-E and why. Especially if this is a tool to be used by game designers, one needs to make sure that the method does not ignore additional descriptors that are a crucial part of the game’s design.

The model **CLIP** [5], which is part of the DALL-E pipeline, can also be used on its own to gauge how well a text description matches an image. We tried matching the images of some original Mario levels to certain prompts that describe a level in more detail, like “under ground Mario level”. For that, we used images of Mario levels provided by the Video Game Level Corpus (VGLC) [7]. The results indicate, that while CLIP does seem to distinguish


DALL·E mini
AI model generating images from any prompt!
Run
DALL·E mini
AI model generating images from any prompt!
Run

A mario level in the style of a apocalyptic hellscape

Spike pit in a Mario level

Figure 1 Examples of images generated with DALL-E Mini [3]. The generated examples look like Mario levels and certain prompts, such as "apocalyptic" can change the style of the generated images. However, prompts like "spike pit" are ignored.

between "over ground" and "under ground" levels, objects such as pipes do not seem to be recognized as well, which likely explains why pipes cannot be easily generated either. Note, that these results are also without fine-tuning the model at all.

The results of these experiments indicate, that even without any fine-tuning of DALL-E and CLIP, the methods already show some understanding of video game levels. We identify creating a small data set of levels and their appropriate textual descriptions for fine-tuning as an important task to further research in this direction.

3.1.3 A Functional Pipeline

While creating images of levels with DALL-E can indicate whether or not the method can be used for level generation in general, this neglects the problem of creating a playable level from that image. Therefore, we established a prototype pipeline for getting playable levels from text, which is shown in Figure 2. Text can be sent to DALL-E to create a level image. Preexisting tools can derive a structured level segment from the image. For now, only the naive RGB tile matching method provided in [2] is applied to this task. Finally, a larger, more complete level can be made from that segment with TOAD-GAN [1]. As the example of a Mario level snippet in Figure 2 shows, the preliminary pipeline works and can create playable Mario level snippets from DALL-E mini. From here, the pipeline needs to be completed by implementing other options to create a tile map from an image, as well as assembling the implemented parts of the pipeline into one cohesive system.

3.1.4 Conclusion and Future Work

In this workgroup, we investigated the possibility of using current text to image methods like DALL-E for video game content generation. We show promising results for Super Mario level generation while identifying problems of the method ignoring certain prompts that

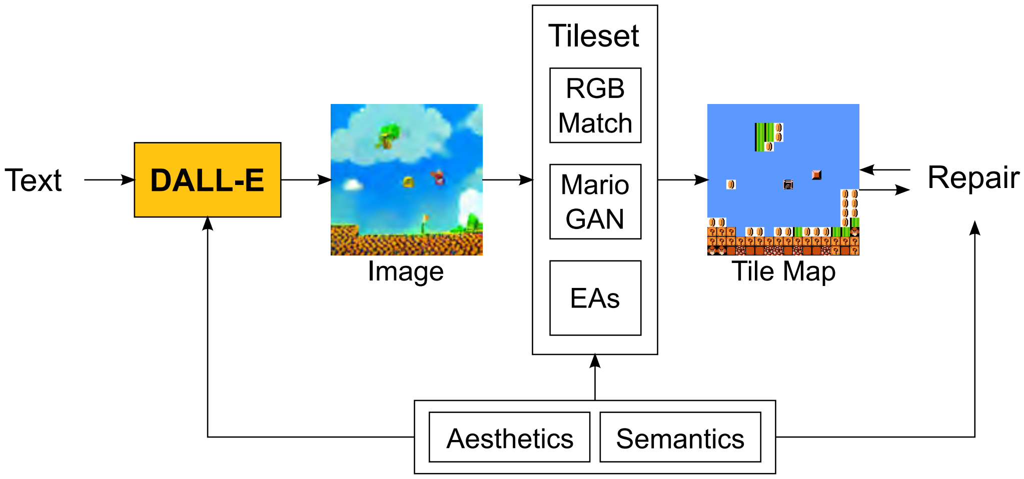


Figure 2 The pipeline established at the end of the group session. As shown with the examples, we implemented the pipeline up to the point of being able to generate a tile map from a text prompt. The image is turned into the tile map with an RGB matching algorithm based on [2].

might be important for a game designer. Additionally, we tested if CLIP, a part of DALL-E, can match certain prompts with given Mario level images, and find a similar result: That it can only distinguish some prompts and might ignore others. This however, is only using the pretrained models as is, and we pose that fine-tuning will improve the results for both experiments. We also established a functional text to level pipeline, which can turn a text prompt into a playable Mario level snippet.

For future work, there are two distinct goals: creating a data set to allow for fine-tuning a pretrained DALL-E and completing the missing pieces of the pipeline. For the data set, detailed descriptions of level images need to be found or created and a way to convert them into a usable format needs to be found. Also, other kinds of data sets that deal with assets other than levels can be explored, like texture images. The missing pieces of the pipeline include include other tile set representations that generate a tile map from an image, such as Generative Adversarial Networks and Evolutionary Algorithms, and using the Tile-Pattern KL-Divergence [4] as a repair mechanism for the tile maps. Also, the currently still fragmented pieces need to be combined to form one cohesive system for ease of use.

References

1 Maren Awiszus, Frederik Schubert, and Bodo Rosenhahn. *Toad-gan: Coherent style level generation from a single example.* In Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 2020

2 Eugene Chen, Christoph Sydora, Brad Burega, Anmol Mahajan, Abdullah Abdullah, Matthew Gallivan, and Matthew Guzdial. *Image-to-level: Generation and repair.* In Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, volume 16, pages 189–195, 2020

3 Boris Dayma, Suraj Patil, Pedro Cuenca, Khalid Saifullah, Tanishq Abraham, Phúc Lê Khac, Luke Melas, and Ritobrata Ghosh. *Dall-e mini*, 7 2021

4 Simon M Lucas and Vanessa Volz. *Tile pattern KL-divergence for analysing and evolving game levels.* In Proceedings of the Genetic and Evolutionary Computation Conference, pages 170–178, 2019

5 Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. *Learning trans-*

ferable visual models from natural language supervision. In Proceedings of the International Conference on Machine Learning, pages 8748–8763. PMLR, 2021

6 Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. *Zero-shot text-to-image generation*. ArXiv preprint, abs/2102.12092, 2021

7 Adam James Summerville, Sam Snodgrass, Michael Mateas, and Santiago Onta n'on Villar. *The vglc: The video game level corpus*. Proceedings of the 7th Workshop on Procedural Content Generation, 2016

3.2 AI for Romantic comedies

Michael Cook (Queen Mary University of London, GB), Maren Awiszus (Leibniz Universität Hannover, DE), Duygu Cakmak (Creative Assembly – Horsham, GB), Alena Denisova (University of York, GB), Alexander Dockhorn (Leibniz Universität Hannover, DE), Casper Harteveld (Northeastern University – Boston, US), Antonios Liapis (University of Malta – Msida, MT), Mirjam Palosaari Eladhari (Stockholm University, SE), Diego Perez Liebana (Queen Mary University of London, GB), Lisa Rombout (Tilburg University, NL), and Tommy Thompson (AI and Games – London, GB)

3.2.1 Introduction and Work Process

Both romance and comedy are integral parts of human culture, yet despite the breadth of AI research into games and creativity, little work has been done to explore these themes in the context of games. In AI research, the best examples are games that deal with 'social physics' or human relationships, such as *Prom Week* [4] or *Façade* [3], where both romantic and comedic themes are hinted at. In the games industry, while romance is a key feature in many games (such as *The Sims*), it is often reduced to static linear narratives, while comedy is notoriously difficult to achieve in games and is often achieved unintentionally [2].

In this workgroup, we aimed to explore the possibility that these two things are connected. Due to a lack of AI research into topics such as romance and comedy, there are fewer systems and techniques available to support the exploration of these themes in game design. Our workgroup aimed to explore the potential for AI research in these areas, to think about the open questions and pitfalls ahead, and to collaboratively sketch out some ideas for work that we could act as inspiring examples for future AI research projects. The group began with a short presentation, including a series of tweets from @NightlingBug on Twitter, who made an observation that playing a game such as *Stardew Valley* from the perspective of a character competing for the player's attention would be an interesting idea.

We began with an open discussion of the topic, encouraging perspectives from everyone present, covering both existing examples of technology and games, as well as concerns, questions, and ideas that arose as we thought about the topic. All of the topics that came out of this discussion were interesting and thought-provoking, but a few ideas stood out as something the groups were particularly excited to take forward during the day. The first was the idea of connecting existing AI narrative techniques, such as the Nemesis system in *Shadow of Mordor* [7] to large-group dynamics like the romantic NPCs in *Stardew Valley*. The second idea was to think about how information flow is often crucial in romantic stories, both within