Personalizable Virtual Experiences

Creating Virtual Reality Content From Text Using Generative Artificial Intelligence

BRIAN LIANG, University of Washington

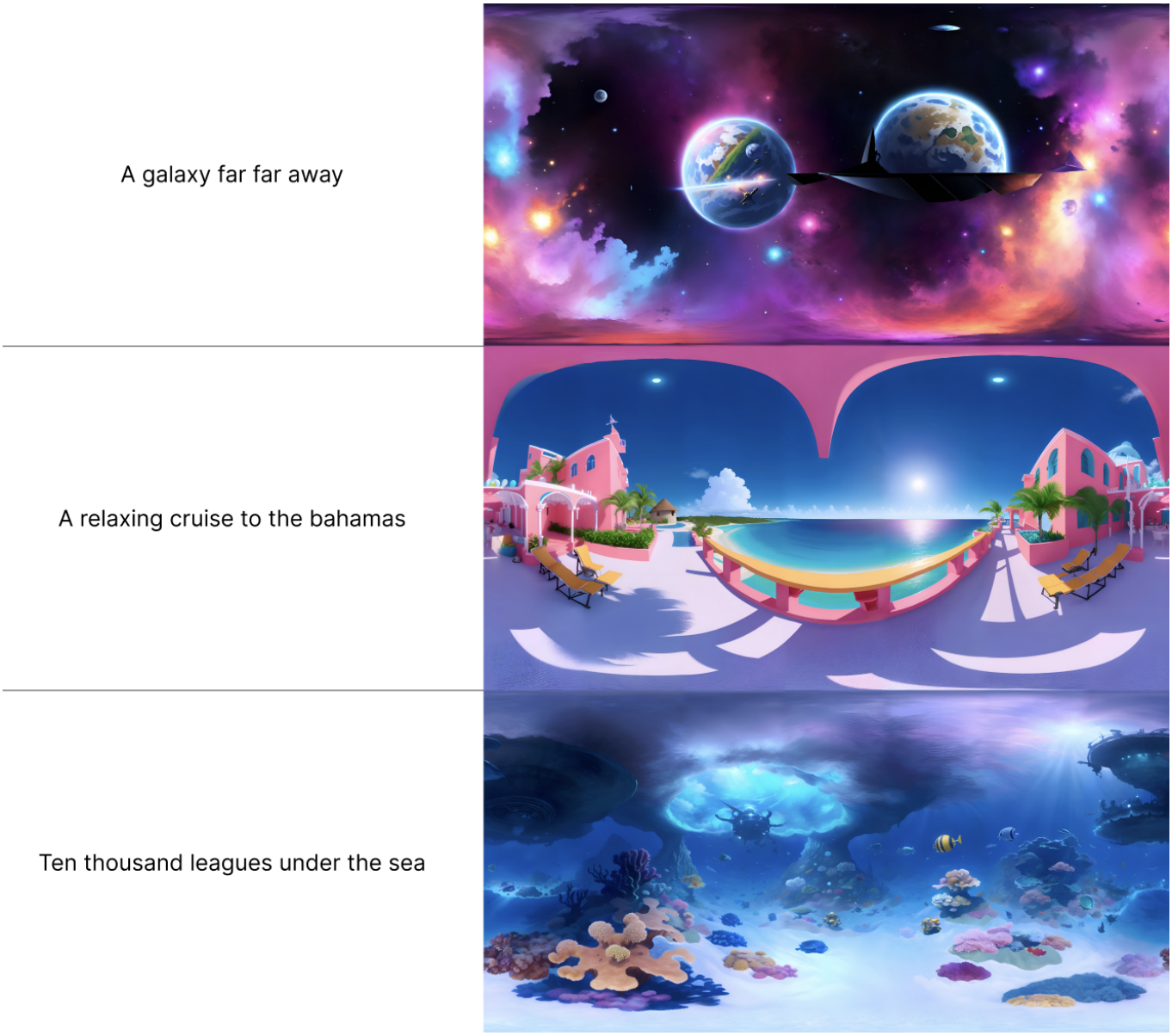

Fig. 1. Generating virtual reality experiences from a line of text.

A major goal of virtual reality systems is to provide users with virtual experiences that can match reality [Terashima 2002]. We aim to generalize these virtual experiences such that virtual reality systems will have the capability to provide users with any experience they want. We develop a system that takes in a text description and generates a virtual reality experience matching the description. The quality of the generated content is empirically reasonable, but far from modern production quality. However, significant portions of our system are active areas of research, meaning that the quality of content generated from our system will improve over time.

Author's address: Brian Liang, liangb1@cs.washington.edu, University of Washington.

1 INTRODUCTION

Many societal challenges can be attributed to one primary issue: a species with an infinitely scaling demand living in a world with finite resources. Creating a digital world without these limitations could potentially mitigate many of these challenges. We approach this goal by using generative artificial intelligence to create virtual reality experiences. Generative AI technologies have already made significant breakthroughs in generating content across many mediums and is constantly improving. While generative AI has the potential to create virtual experiences, no complete end-to-end system has been developed for this purpose yet.

In this area, prior works have mostly investigated the development of assets that make up a digital experience. Complete end-to-end systems have been investigated, but not as thoroughly. Additionally, current state-of-the-art implementations of end-to-end generative experiences lack interactive elements, a major component of realistic virtual experiences. Our approach addresses weaknesses in current systems using state-of-the-art generative models. We discuss these works in section 2.

We aim to create a complete end-to-end system that creates a virtual reality experience from a given text description. We use large language models to predict components that make up the experience, convert these into graphics using text to 3D models, add interactions to components with large language models, and compose all of the elements together to complete the experience. We discuss our methods in section 3 and implementation in section 4.

Our system succeeds in creating functioning virtual experiences that match the input description with reasonable accuracy, but lacks quality compared to modern production. We discuss our system evaluation in section 5. The results of our investigation suggest a bright future for generative virtual experiences. We discuss potential areas of future work in section 6.

Contributions

- We propose and motivate the new problem of generalizing virtual experiences.
- We developed the first functional end-to-end text to virtual experience system.
- We thoroughly evaluate our approach and formulate strategies on how it can be perfected.

2 RELATED WORK

Generative AI Models

The task of creating digital content spans several different mediums. Large language models are applicable for this task in creating textual assets and generating code. Generative image models are useful for creating graphical assets. Due to the massive training costs for these models, it is often best practice to use models pre-trained by institutions with access to large computational resources. Our approach uses the large language model GPT-3 created by Open AI [Brown et al. 2020] and the generative image model Stable Diffusion created by Stability AI [Rombach et al. 2022].

Generating Digital Assets

The task of generating digital assets has received much attention in recent years . Advancements have been made in generating 3D assets, notably Get3D [Gao et al. 2022], DreamFusion [Poole et al. 2022], and Shap-E [Jun and Nichol 2023]. Furthermore Inworld AI has used language models to control character interactions. Our approach uses the text to 3D model Shap-E created by Open AI [Jun and Nichol 2023].

End-to-End Systems

Developments in end to end systems have received a good amount of attention as well. The task of generating 3D scenes has been investigated in Set-the-Scene [Cohen-Bar et al. 2023], SceneScape [Fridman et al. 2023], and Text2Room [Höllein et al. 2023]. Furthermore, Blockade Labs has created a system to project generated images onto a panoramic cylinder to enable 360° view of the image from a virtual reality headset. One major weakness is that developments in generating interactions along with scenes have been limited. We aim to address this weakness by developing a complete end-to-end system with interactions.

3 METHOD

Addressing the challenge of generalizing virtual experiences may seem unachievable. We attempt to take a step in the right direction with a chain generation and composition approach. We begin by taking in as input a text description of a digital experience from the user. The user shouldn't have to write too much, but the system also needs to have enough details to be able to create the experience. We generate these details by feeding the text description through a language model. From this, we obtain viable descriptions of elements that make up the experience. We then take these descriptions and use a text to 3D model to create graphics for each element. The graphics are then composed into a 3D scene. Elements in the scene are given interactive attributes from their descriptions using language models. The composed scene can then be converted into a virtual reality experience.

To gain some intuition behind the system, we can begin by making the assumption that for all experiences, there is a way to represent each one digitally. This must be true because any digital experience is just some combination of pixels, and every combination of pixels can be shown on a screen. We use large generative models as models of the world; because they are built on massive quantities of data, we can make the assumption that they accurately represent a distribution of the digital world. Through prompting, we can then sample the parts of this distribution that represent the user's description, and convert this sample into a 3D representation to create a virtual experience matching the description.

4 IMPLEMENTATION DETAILS

We implement our system in Python. We begin by taking a user defined input string. This string is added to a prompt querying more details about the input. The prompt is sent to GPT-3, and we receive details in our specified format that we parse and process. For each component, we query Shap-E with the component's graphical description, and receive a mesh of the component. We then query

GPT-3 with the details of the component to receive information regarding the dynamics of the component (if applicable). When the component is interacted with, it will query GPT-3 with the text from the interaction along with a description of the component and respond with a GPT-3 generated response that accurately matches the characteristics of the component. To compose the scene, we generate an image of the scene using Stable Diffusion to create a layout of components. We run object detection on the image using YOLO and single image depth estimation using MiDaS. We convert these estimates to coordinates in a 3D space, and place each component in the scene in the position such that we have the best label cosine similarity between the detected objects and component list. We then render and simulate our scene in pybullet.

5 EVALUATION AND DISCUSSION OF RESULTS

Our results were reasonable, but left a lot to be desired. The generated graphics were low quality due to both the limitations of the model and to our limited computational resources. The model generated 64x64x64 pixel meshes, which often contained deformities. Dynamics were also challenging to simulate. The components would move around with seemingly random behaviors, instead of moving cohesively to push the plot of the generated experience. Scene layouts were also not the best, due to deformities in generated images. Given additional resources, a better way to generate 3D scenes would be to use data driven methods. In general, the textual aspects of the system worked well. Generated intermediary descriptions were accurate to the inputs. Generated responses from components were always accurate to the context of the interaction. Because of this, our system was able to generalize fairly well. One of the reasons our system may have had challenges with quality but performed well in generalization is because using large AI models breaks the assumption that we have a perfect model of the world; generative AI models by nature will make mistakes at the expense of generalizing well.

6 FUTURE WORK

The biggest bottleneck in our system is the quality of 3D scene generation. While recent works have made improvements in this area, there is a lack of 3D data relative to the text and image data that is available. Once 3D generation catches up to text and image generation, this system will perform much better.

7 CONCLUSION

In conclusion, we developed a system that generates a virtual reality experience from a given input text. While we were able to generalize these experiences fairly well to a wide range of inputs, the quality of these experiences left a lot to be desired. The future for generalizable virtual reality experiences looks bright.

ACKNOWLEDGMENTS

Huge thank you to Douglas Lanman and the UW Reality Lab for teaching me about virtual reality systems and providing resources to help me with this project!

REFERENCES

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. arXiv:2005.14165 [cs.CL]

Dana Cohen-Bar, Elad Richardson, Gal Metzer, Raja Giryes, and Daniel Cohen-Or. 2023. Set-the-Scene: Global-Local Training for Generating Controllable NeRF Scenes.

Rafail Fridman, Amit Abecasis, Yoni Kasten, and Tali Dekel. 2023. SceneScape: Text-Driven Consistent Scene Generation. arXiv:2302.01133 [cs.CV]

Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. 2022. GET3D: A Generative Model of High Quality 3D Textured Shapes Learned from Images. arXiv:2209.11163 [cs.CV]

Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. 2023. Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models. arXiv:2303.11989 [cs.CV]

Heewoo Jun and Alex Nichol. 2023. Shap-E: Generating Conditional 3D Implicit Functions. arXiv:2305.02463 [cs.CV]

Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. 2022. DreamFusion: Text-to-3D using 2D Diffusion. arXiv:2209.14988 [cs.CV]

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-Resolution Image Synthesis with Latent Diffusion Models. arXiv:2112.10752 [cs.CV]

Nobuyoshi Terashima. 2002. 11 - Telesensation. In *Intelligent Communication Systems*, Nobuyoshi Terashima (Ed.). Academic Press, San Diego, 127–148. https://doi.org/10.1016/B978-012685351-3/50012-3