Out-of-distribution Shape Generation using Large Language Models and Geometry Nodes

Yutaro Yamada[1], Machel Reid[2]
[1]Yale University [2]Google DeepMind
yutaro.yamada@yale.edu

概要

Recent advancements in text-to-image models have been impressive. Yet, they still struggle with accurately following prompts, especially when asked to create out-of-distribution objects like a "chair with three legs." In this paper, we introduce a novel method using "Geometry Nodes"–a procedural generation tool for creating shapes in Blender, which is a 3D modeling software. This tool allows for the generation of shapes based on general parameters like 'radius' and 'height.' We use a large language model (LLM) to interpret text prompts and convert them into specific instructions for these shape generators. Our method has proven effective in creating unconventional examples in two categories: Chairs and Tables.

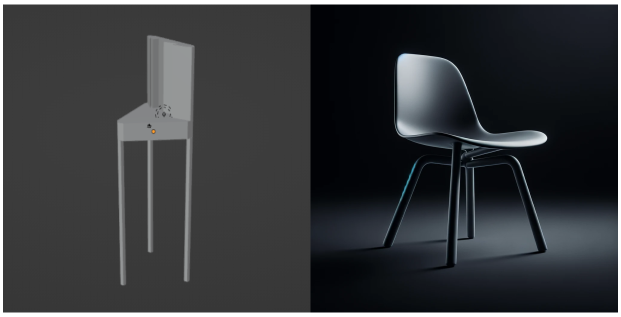

図 1 Our approach vs. DALL-E-3, one of the most advanced text-to-image models for the task of generating out-of-distribution samples like "a chair with three legs". For DALL-E-3, we use "A chair with three legs. Make sure the background is black, and the object generated is a gray-colored 3D shape." for our prompt. For our approach, our prompt is "a chair with three legs."

1 はじめに

Models like DALL-E-3 [1] and Stable Diffusion [2] have made significant advancements in text-to-image generation, producing images nearly indistinguishable from those created by humans. However, they still face challenges in accurately following prompts, particularly in representing spatial relationships and object-attribute binding. Various solutions have been suggested to overcome these issues. For example, [3] propose to enhance pre-trained diffusion models with additional inputs like scribbles or depth maps. [4] uses compositional linguistic structures during the diffusion guidance.

While these methods show promise, they still struggle with generating images of objects that do not typically exist, like a "a chair with five legs." This difficulty partly arises because such unusual objects are rarely included in the training data. Additionally, it is likely that these models do not separate and understand concepts the way humans do, such as distinguishing the number of legs from the general concept of a chair.

Large Language Models (LLMs) are known for their exceptional reasoning abilities [5]. This raises the question: can we use the reasoning capabilities of LLMs to solve the above problem? Our approach involves using Blender's Geometry Nodes, a tool to create parametric shape generator, in combination with the function-calling ability of LLMs. By feeding specific arguments to the shape generator, we can partially address the above problem.

Geometry Nodes is a tool within Blender, an open-source 3D modeling software, that enables the creation of 3D meshes programmatically. It allows users to build a computational graph of 3D operations and set certain parameters as inputs to this graph. For instance, using a set of Geometry Nodes, one can create various designs of objects by altering the input parameters. An example illustration of Geometry Nodes setup is shown in Figure 3. This flexible system is especially popular among programmer-oriented 3D designers, who use Geometry Nodes to instantly generate diverse shapes by tweaking specific parameters.

In our work, we extend this concept. We have developed two parametric shape generators for chairs and tables. Uniquely, these generators allow for the modification of certain attributes, like the number of legs. Alongside these generators, we provide docstrings - natural language descriptions explaining how to use each shape generator effectively.

2 Related Work

Alignment with human intentions for text-to-image models The topic of aligning text-to-image models with human intentions has been gaining significant interest. Recently, there have been various methods to ensure that image generation aligns more closely with what humans intend. One effective approach involves using additional inputs like layout [6], bounding boxes [7], and user sketches [3] to steer the direction of image generation.

Text-guided image editing is another method that incorporates human intentions. Recent advancements include using pre-trained diffusion models along with a spatial mask to focus edits in specific areas, as explored by [8]. Another approach uses cross-attention layers for text-only guidance in image editing, as demonstrated in [9]. Imagic, introduced by [10], stands out for its ability to make complex, non-rigid semantic edits through text-based optimization. The concept of model customization , as explored by [11, 12], is particularly noteworthy. These techniques allow diffusion models to create images of a new concept from just a handful examples. Once the model is fine-tuned to understand the concept, it can generate various images of that subject in different contexts.

Prompt following There is ongoing work to improve how diffusion models interpret and accurately follow text prompts. Early versions of Stable Diffusion faced challenges with correctly associating attributes and generating compositions. For instance, they often misassigned colors in images with two concepts or struggled to generate images with multiple concepts. [4] tackles this problem by integrating linguistic structures into the diffusion process. 'Attend-and-excite,' proposed by [13], employs an attention-based semantic guidance method. DALL-E-3 [1] largely resolves these issues by using better, more detailed captions on a larger scale.

Procedural generation In the realm of procedural generation, Geometry Nodes have gained traction in the 3D vision community. For instance, InfiniGen [14] employs Geometry Node-based methods to create 3D scenes, providing an endless source of synthetic data for 2D and 3D vision tasks. There is also a work [15] that combines InfiniGen with Large Language Models (LLMs), using LLMs as a language interface to control world creation. Our method is distinct; we introduce new shape generators based on Geometry Nodes, focusing specifically on creating samples that are outside the usual training distribution.

3 Method

Our proposed text-to-3D approach involves LLM's function calling ability and shape generator based on Geometry nodes. The overall approach is shown in Figure 2. We describe each step below.

3.1 LLM function calling

The first step in our process is to translate a user's natural language description into a set of parameters for the shape generator. To do this, we utilize the function calling capability of Large Language Models (LLMs). This feature, originally developed by OpenAI, enhances LLMs' chat completion abilities, enabling them to connect seamlessly with external tools and applications. Many external applications offer their functionalities through APIs (Application Programming Interfaces). The ability of LLMs to execute these APIs in response to user text inputs is a natural extension of their capabilities. While it is possible to prompt LLMs to generate strings in a specific format, it is more efficient for them to directly produce outputs in JSON format. To meet this requirement, LLMs can be fine-tuned to output in these specific formats. This approach has been gaining popularity, and there are efforts within the open-source community, like the one led by [16], to replicate OpenAI's function calling feature. To enhance the interaction between LLMs and these applications, we provide detailed natural language descriptions of what each argument represents.

3.2 Parametric shape generator based on Geometry Nodes

The second phase in our text-to-3D process involves a parametric shape generator using Geometry Nodes. In Blender, each Geometry Node corresponds to a specific operation, like creating a mesh, extruding mesh surfaces, and selecting certain mesh faces through various selection

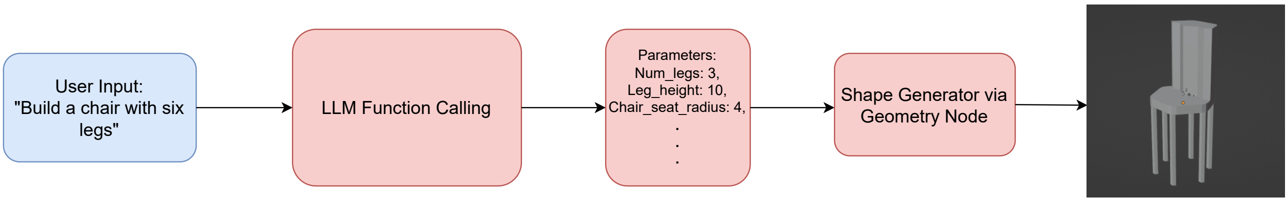

図 2 Our text-to-3D pipeline using Large Language Models and Geometry Nodes.

mechanisms. In our project, we have developed two specialized Geometry Node groups specifically for chairs and tables. These groups are designed to be flexible, allowing input parameters to adjust features such as the number of legs, height, width, and curvature of the furniture. An example of our Geometry Node setup are displayed in Figure 3. The arguments to the chair generator are summarized as follows: "number of legs", "leg height", "leg radius", "chair seat height", "chair seat radius", "chair seat corner radius", "Distance to legs from chair seat edge", "backrest height". The arguments for the table generator are summarized as follows: "number of legs", "leg height", "leg radius", "tabletop height", "tabletop radius", "Distance to legs from table edge". We then convert our Geometry Node groups into Python scripts by using a converter [1]. This allows us to take the arguments we receive from the function calling and pass them directly to the scripted version of the Geometry Nodes.

4 Results

Here we showcase how our text-to-3D pipeline works for generating regular chairs and tables via user's text input. We then show how we can also generate out-of-distribution chairs and tables, and compare results from DALL-E-3 and StableDiffusion-XL.

4.1 Text-guided generation

We first test our text-to-3D pipeline to generate regular chairs and tables. As shown in Figure 4, we can see that the height attribute in text prompts is correctly reflected in generated 3D mesh.

4.2 Out-of-distribution sample generation

Next we compare our approach with some of the most advanced text-to-image models: DALL-E-3 and Stable Diffusion XL. The results are shown in Figure 5 and Figure 6. We see that DALL-E-3 fail to accurately follow the input prompts. SDXL successfully generates "a table with three legs" and "a chair with five legs" (if we consider one of the overlapping legs as the fifth leg.) However, for the rest of cases, we see that SDXL struggle to follow the text prompts. On the other hand, our approach successfully generates desired shapes in all cases. This shows that the reasoning ability of LLMs along with parametric shape generator as a structured prior helps in the case of generating out-of-distribution samples.

5 Limitations

Our approach requires parametric shape generators, which limits the applicability of our method. Future work should explore if it is possible to learn to generate these parametric shape generators, which can be written as a code.

6 Conclusion

We tackle the issue of prompt following of text-to-image and text-to-3D models. We first show that even the most advanced text-to-image models still struggle with generating out-of-distribution samples. We then introduce our approach that combines Large Language Models and Geometry Nodes. Large Language Models deal with reasoning and translating user prompts into concrete API calls, which then be used to execute Geometry Nodes to create 3D mesh. We show that our approach can generate both regular four-legged chairs as well as out-of-distribution chairs like "a chair with three legs" and "a chair with six legs". Since our approach requires preparing Geomtry Node setups prior to generation, this limits the applicability of our method to pre-defined object categories. However, our approach shows greater flexibility and control over the range of characteristics that are beyond commonly seen attributes in training data, thanks to the reasoning ability of Large Language Models.

[1] https://github.com/BrendanParmer/NodeToPython

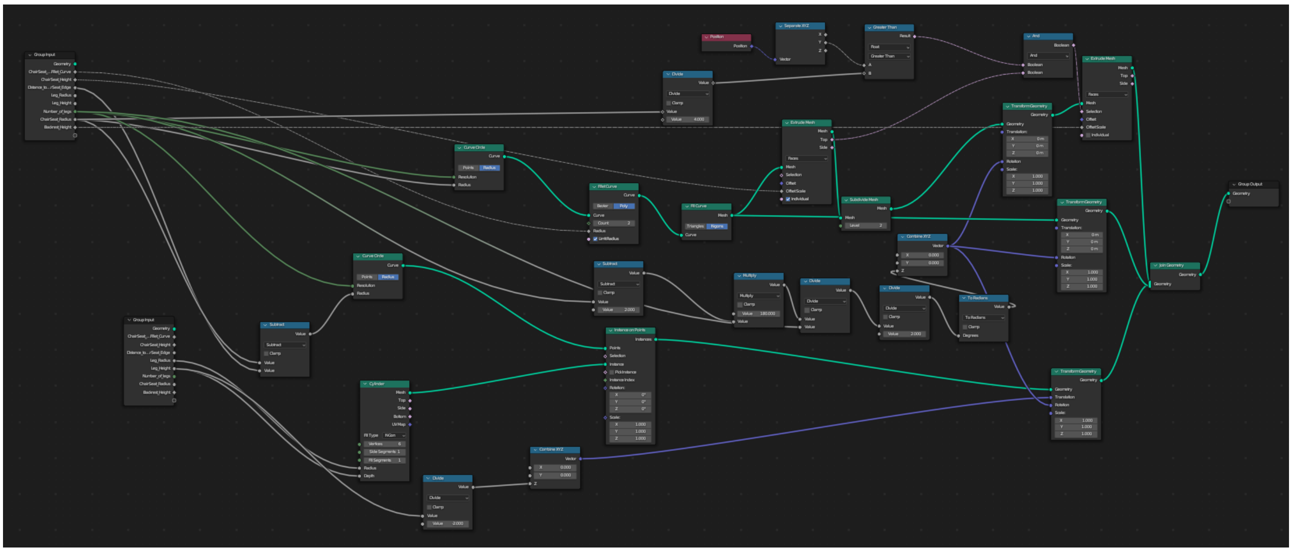

図 3　An example geometry node setup. Each box represents a mesh operation such as “Extrude mesh”, “Create a curve circle” etc.

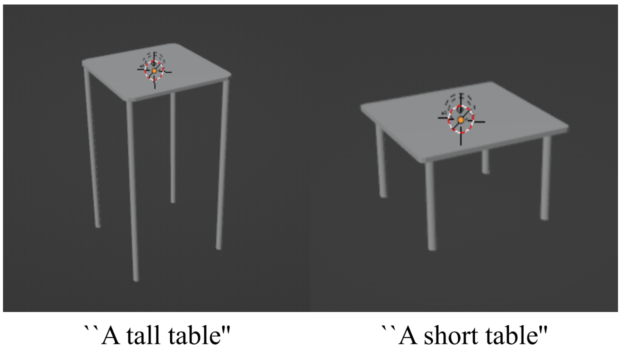

図 4　An example 3D mesh result for tables.

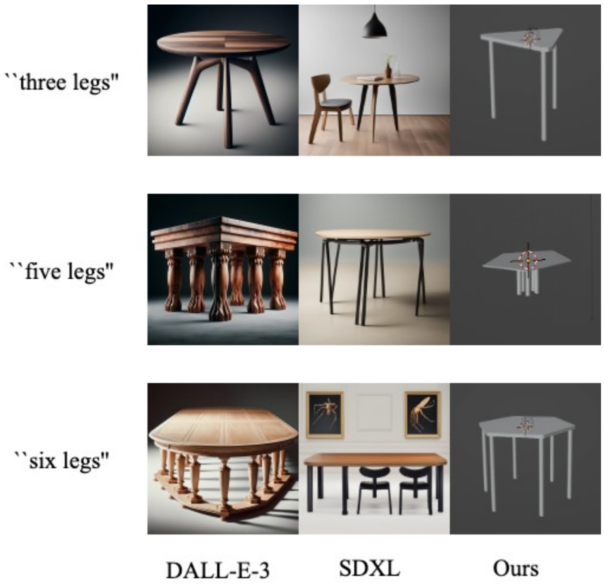

図 5　Comparing DALL-E-3, Stable Diffusion-XL (SDXL), and our approach for generating out-of-distribution tables. The prompts we use are: “a table with three legs/five legs/ six legs”, respectively.

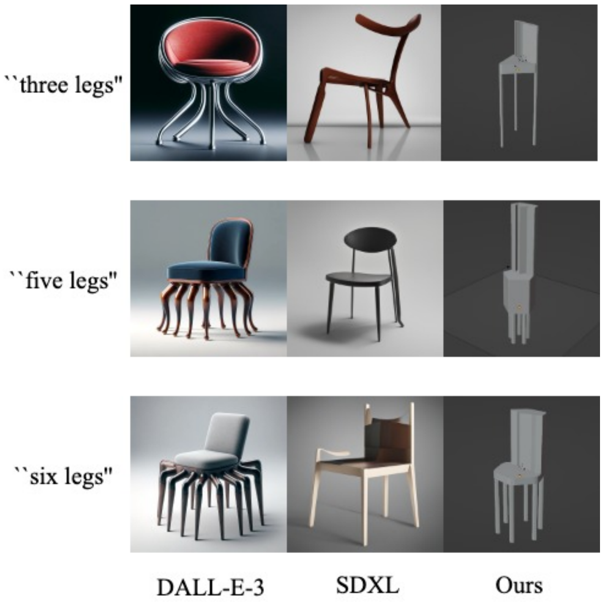

図 6　Comparing DALL-E-3, Stable Diffusion-XL (SDXL), and our approach for generating out-of-distribution chairs. The prompts we use are: “a chair with three legs/five legs/ six legs”, respectively.

参考文献

[1] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, Wesam Manassra, Prafulla Dhariwal, Casey Chu, Yunxin Jiao, and Aditya Ramesh. Improving Image Generation with Better Captions.

[2] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis, July 2023.

[3] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding Conditional Control to Text-to-Image Diffusion Models. In

Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3836–3847, 2023.

[4] Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Reddy Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-Free Structured Diffusion Guidance for Compositional Text-to-Image Synthesis. In **The Eleventh International Conference on Learning Representations**, September 2022.

[5] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. ReAct: Synergizing Reasoning and Acting in Language Models. In **The Eleventh International Conference on Learning Representations**, September 2022.

[6] Zhengyuan Yang, Jianfeng Wang, Zhe Gan, Linjie Li, Kevin Lin, Chenfei Wu, Nan Duan, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. ReCo: Region-Controlled Text-to-Image Generation. In **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**, pp. 14246–14255, 2023.

[7] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. GLIGEN: Open-Set Grounded Text-to-Image Generation. In **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**, pp. 22511–22521, 2023.

[8] Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended Latent Diffusion. **ACM Transactions on Graphics**, Vol. 42, No. 4, pp. 149:1–149:11, July 2023.

[9] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-Prompt Image Editing with Cross-Attention Control. In **The Eleventh International Conference on Learning Representations**, September 2022.

[10] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-Based Real Image Editing With Diffusion Models. In **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**, pp. 6007–6017, 2023.

[11] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation. In **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**, pp. 22500–22510, 2023.

[12] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-Concept Customization of Text-to-Image Diffusion. In **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**, pp. 1931–1941, 2023.

[13] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-Excite: Attention-Based Semantic Guidance for Text-to-Image Diffusion Models. **ACM Transactions on Graphics**, Vol. 42, No. 4, pp. 148:1–148:10, July 2023.

[14] Alexander Raistrick, Lahav Lipson, Zeyu Ma, Lingjie Mei, Mingzhe Wang, Yiming Zuo, Karhan Kayan, Hongyu Wen, Beining Han, Yihan Wang, Alejandro Newell, Hei Law, Ankit Goyal, Kaiyu Yang, and Jia Deng. Infinite Photorealistic Worlds Using Procedural Generation. In **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**, pp. 12630–12641, 2023.

[15] 3D-GPT: Procedural 3D Modeling with Large Language Models. In **The Twelfth International Conference on Learning Representations**, October 2023.

[16] Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. Gorilla: Large Language Model Connected with Massive APIs, May 2023.