Generative 3D Mesh Modeling with Text-to-Texture Generator

JIALUO LI* and ZIRU HUANG

Our project presents an extension to the MeshDiffusion model [Liu et al. 2023], incorporating class conditioning to enable the generation of 3D meshes based on input class labels (5 categories are availiable). Additionally, we employ a pre-trained 2D diffusion model to distill knowledge, align textures with input textual descriptions. This integration results in an effective method for imparting textures onto 3D meshes, facilitating a seamless connection between class-based mesh generation and texture synthesis.

Additional Key Words and Phrases: 3D Mesh Generation, Diffusion Model, Class Conditioning, Texture Synthesis, 2D Distillation, MeshDiffusion, Generative Models, Computer Graphics.

ACM Reference Format:
Jialuo Li* and Ziru Huang. 2024. Generative 3D Mesh Modeling with Text-to-Texture Generator. *ACM Trans. Graph.* 1, 1 (August 2024), 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Building upon the MeshDiffusion framework, we introduce an extension that incorporates class conditioning, allowing users to influence the generated 3D meshes based on specified class labels. The inclusion of class information enhances the model's versatility, enabling the synthesis of meshes tailored to specific categories.

Furthermore, to complement the geometric details, we propose the integration of a pre-trained 2D diffusion model utilizing the knowledge distillation technique [Gao et al. 2022]. This secondary model is employed to align textures with textual descriptions, providing a means to seamlessly associate visual details with input attributes. The combination of class-conditioned 3D mesh generation and texture synthesis through 2D diffusion provides a method to generate textured mesh.

In this report, we present the details of our extended MeshDiffusion model, the incorporation of class conditioning, and the integration of a 2D diffusion model for texture alignment. We demonstrate the effectiveness of our approach through experimental results, showcasing the model's ability to generate diverse and category-specific 3D meshes with corresponding textures.

2 METHODOLOGY

2.1 Mesh Parameterization

In order to parameterize the mesh into the neural network, we have adopted the Deep Marching Tetrahedra (DMTET) framework [Shen et al. 2021]. By this framework, the mesh is represented using an SDF encoded with a deformable tetrahedral grid (See 1). Every grid vertex stores 3D offset (difference from the initialization) and its SDF value. In addition, we augment tetrahedral grids to cubic grids for better extracting features from 3D CNN layers.

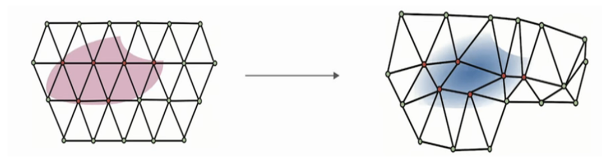

Fig. 1. Mesh Representation Using Deformable Grids

2.2 Class-Conditioned Diffusion Model

Our project builds upon the general framework established by the Latent Diffusion Model [Rombach et al. 2022], as illustrated in Figure 2. We have adapted this architecture by incorporating class embeddings and classifier guidance, enhancing its capabilities for our specific application [Dhariwal and Nichol 2021].

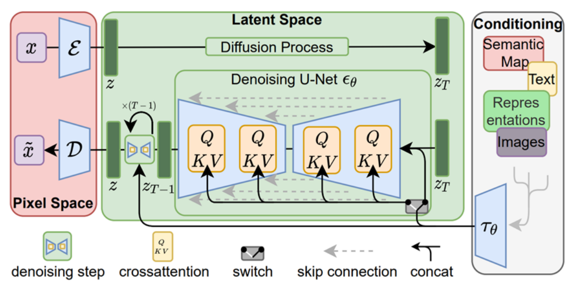

Fig. 2. Latent Diffusion Model Structure

2.2.1 *Class Embedding.*

- Concatenate the input of Resnet block and the pre-defined class embedding
- Use Cross-Attention block to extract the features (See 3).

2.2.2 *Classifier Guidance.*

Furthermore, we employ classifier guidance to refine the generation process. This approach involves training an auxiliary classifier, leveraging the encoder component of the Unet architecture as its foundation [Dhariwal and Nichol 2021].

2.3 2D Distillation Based On Pretrained Diffusion Model

We have integrated the Stable Diffusion model, drawing inspiration from the DreamFusion framework [Poole et al. 2022]. The process begins with the input mesh, which is rendered alongside a trainable texture. This composite is then fed into the diffusion model. The optimization of the texture is guided by a loss function that quantifies the discrepancy between the predicted noise from the diffusion model and the ground truth data (See 4).

Authors' address: Jialuo Li*; Ziru Huang.

0730-0301/2024/8-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

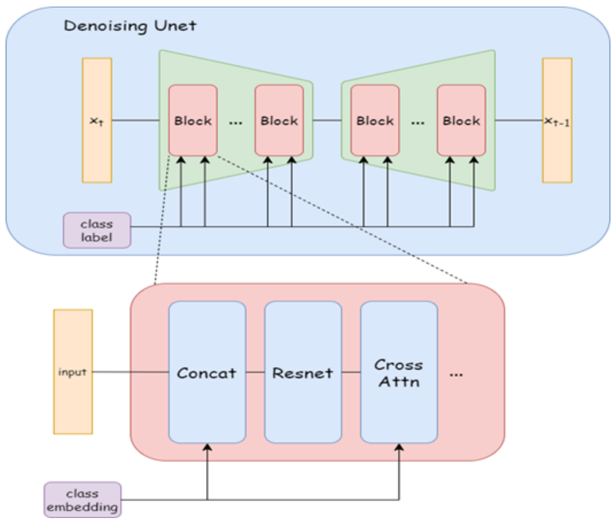

Fig. 3. The Detailed Structure Of Class-Conditioned Model

2.4 Result

2.4.1 Experiment Details. In our training dataset for the class conditioned diffusion model, we have selected five specific categories from ShapeNet [Chang et al. 2015]: planes, tables, chairs, rifles, and cars.

Through our experimental observations, we have determined that initializing the texture generation process with data derived from stable diffusion outperforms random initialization. This is because the former method leads to a more rapid convergence.

Furthermore, we have explored an alternative mesh representation through voxels, utilizing the marching cubes algorithm [Lorensen and Cline 1987]. The outcomes of this approach are depicted in the subsequent figures.

2.4.2 Generated Results. Here, we present the resulting figures from our experiments, as illustrated in figure 5, 6, 7.

Fig. 5. The Generated Mesh With Voxel Based

3 DISCUSSION AND LIMITATIONS

3.1 Main Challenges

3.1.1 Environment Configuration. Our primary implementation relies on the codebase from MeshDiffusion. Unfortunately, the GitHub repository associated with this paper provided minimal guidance on environment configuration, which led to a week-long struggle to set up the necessary environment. This process was further complicated by numerous bugs encountered during the attempt to reproduce the results.

3.1.2 Data Preprocess. Before initiating the training process for our model, it's essential to convert the raw mesh data (in .obj format) into the DMTet format. However, this preprocessing step, which can take between 20 to 30 minutes for a single mesh, has led us to reconsider our approach. The sheer volume of preprocessed data required makes our idea of generating meshes from text inputs impractical.

3.1.3 Computing Resource. Consider the constraints of our computing resources, which necessitate approximately three days for each checkpoint training, we face limitations in conducting extensive experimentation to identify the optimal checkpoint within our time constraints. Consequently, we have adopted a strategy of periodically assessing the model's performance to ensure progress and make informed decisions.

3.2 Limitations

In fact, the quality of our generated outputs is not entirely satisfactory. The meshes produced often exhibit numerous cavities, which can compromise their structural integrity. Additionally, the text-to-texture generator faces challenges when converging to a stable output, particularly when the input text is complex. These issues highlight areas where further refinement and optimization are needed to improve the reliability and consistency of our model's performance.

4 CONTRIBUTION STATEMENT

In our project, I took on the role of developing the texture generation code, leveraging the MeshDiffusion codebase as a foundation. Additionally, I crafted an interactive interface for immersive omnidirectional viewing of the generated meshes. I was also instrumental in conducting the majority of the experiments to validate our results. Our GitHub repository, which houses our project, can be found here, and for a more comprehensive understanding, our project website is available here.

REFERENCES

Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. 2015. ShapeNet: An Information-Rich 3D Model Repository. arXiv:1512.03012 [cs.GR]

Prafulla Dhariwal and Alex Nichol. 2021. Diffusion Models Beat GANs on Image Synthesis. arXiv:2105.05233 [cs.LG]

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. In *Advances In Neural Information Processing Systems*.

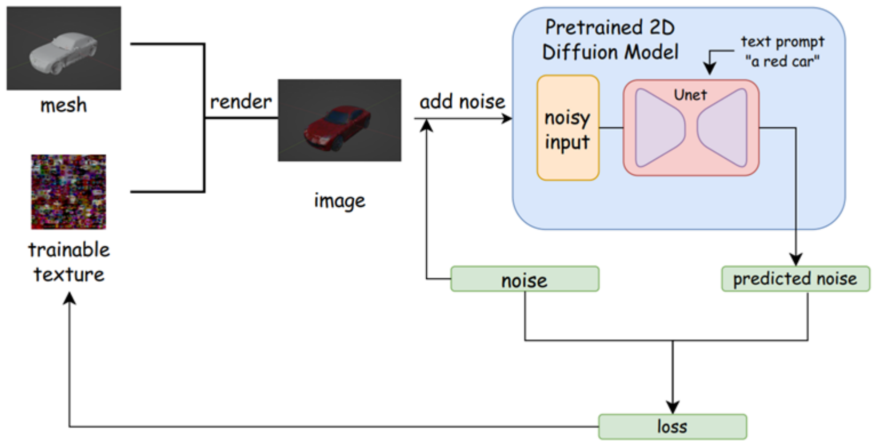

Fig. 4. The Structure of Texture Generator

Fig. 6. The Generated Mesh With Class-Condition

Fig. 7. The Generated Textured Mesh With Text Input

Zhen Liu, Yao Feng, Michael J. Black, Derek Nowrouzezahrai, Liam Paull, and Weiyang Liu. 2023. MeshDiffusion: Score-based Generative 3D Mesh Modeling. arXiv:2303.08133 [cs.GR]

William E. Lorensen and Harvey E. Cline. 1987. Marching Cubes: A High Resolution 3D Surface Construction Algorithm. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '87)*. ACM, New York, NY, USA, 163–169. https://doi.org/10.1145/37401.37422

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]

Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. 2021. Deep Marching Tetrahedra: a Hybrid Representation for High-Resolution 3D Shape Synthesis. arXiv:2111.04276 [cs.CV]