Unique3D: High-Quality and Efficient 3D Mesh Generation from a Single Image

Kailu Wu
Tsinghua University

Fangfu Liu
Tsinghua University

Zhihan Cai
Tsinghua University

Runjie Yan
Tsinghua University

Hanyang Wang
Tsinghua University

Yating Hu
AVAR Inc.

Yueqi Duan[†]
Tsinghua University

Kaisheng Ma[†]
Tsinghua University

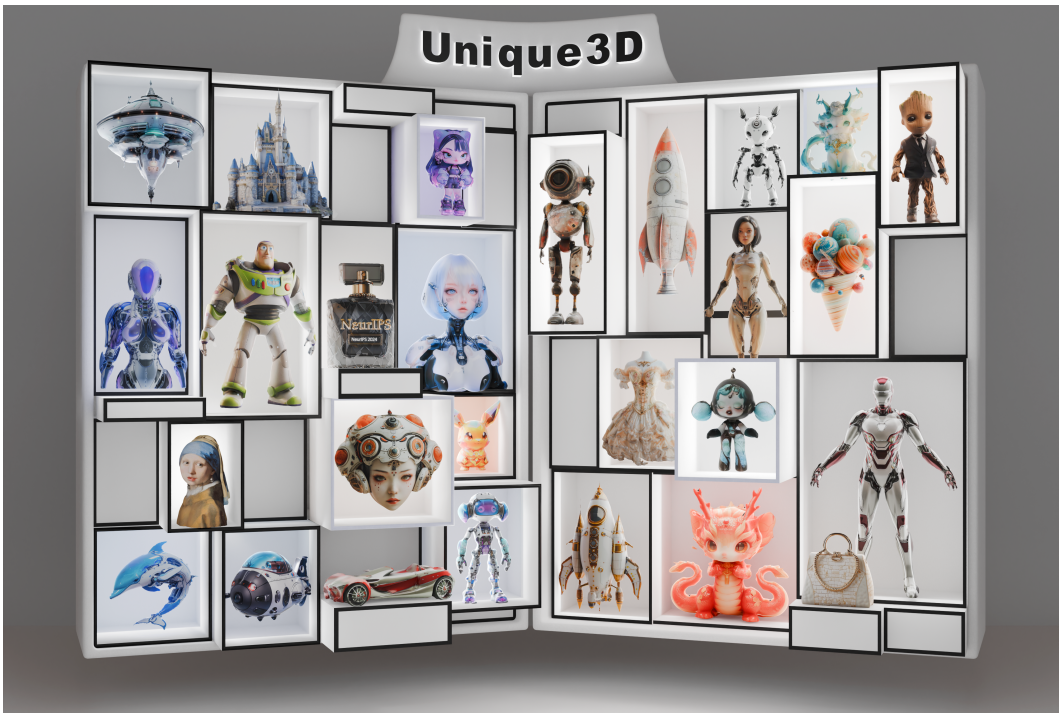

Figure 1: **Gallery of Unique3D**. High-fidelity and diverse textured mesh generated by Unique3D from single-view wild images within 30 seconds.

Abstract

In this work, we introduce **Unique3D**, a novel image-to-3D framework for efficiently generating high-quality 3D meshes from single-view images, featuring state-of-the-art generation fidelity and strong generalizability. Previous methods based on Score Distillation Sampling (SDS) can produce diversified 3D results by distilling 3D knowledge from large 2D diffusion model, but they usually suffer from long per-case optimization time with inconsistent issues. Recent works address the problem and generate better 3D results either by finetuning a multi-view diffusion model or training a fast feed-forward model. However, they still lack intricate textures and complex geometries due to inconsistency and limited gener-

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

ated resolution. To simultaneously achieve high fidelity, consistency, and efficiency in single image-to-3D, we propose a novel framework Unique3D that includes a multi-view diffusion model with a corresponding normal diffusion model to generate multi-view images with their normal maps, a multi-level upscale process to progressively improve the resolution of generated orthographic multi-views, as well as an instant and consistent mesh reconstruction algorithm called *ISOMER*, which fully integrates the color and geometric priors into mesh results. Extensive experiments demonstrate that our Unique3D significantly outperforms other image-to-3D baselines in terms of geometric and textural details. Project page: `https://wukailu.github.io/Unique3D/`.

1 Introduction

Automatically generating diverse and high-quality 3D content from single-view images is a fundamental task in 3D Computer Vision [15, 13, 29, 60, 63], which can facilitate a wide range of versatile applications [25, 49], including gaming, architecture, art, and animation. However, this task is challenging and ill-posed due to the underlying ambiguity of 3D geometry in a single view.

Recently, the rapid development of diffusion models [12, 52, 39] has opened up new perspectives for 3D content creation. Powered by the strong prior of 2D image diffusion models, DreamFusion [36] proposes Score Distillation Sampling (SDS) to address the limitation of 3D data by distilling 3D knowledge from 2D diffusions [41], inspiring the progress of SDS-based 2D lifting methods [20, 37, 59, 24, 5]. Despite their diversified compelling results, they usually suffer from long per-case optimization time for hours, poor geometry, and inconsistent issues (*e.g.,* , Janus problem [36]), thus not practical for real-world applications. To overcome the problems, a series of works leverage larger-scale open-world 3D datasets [7, 4, 6] either to fine-tune a multi-view diffusion model [29, 28, 57] and recover the 3D shapes from the generated multi-view images or train a large reconstruction model (LRM) [13, 64, 60, 63] by directly mapping image tokens into 3D representations (*e.g.,* , triplane or 3D Gaussian [16]). However, due to local inconsistency in mesh optimization [29, 65] and limited resolution of the generative process with expensive computational overhead [13, 63], they struggle to produce intricate textures and complex geometric details with high resolution.

In this paper, we present a novel image-to-3D framework for efficient 3D mesh generation, coined **Unique3D**, to address the above challenges and simultaneously achieve high-fidelity, consistency, and generalizability. Given an input image, Unique3D first generates orthographic multi-view images from a multi-view diffusion model. Then we introduce a multi-level upscale strategy to progressively improve the resolution of generated multi-view images with their corresponding normal maps from a normal diffusion model. Finally, we propose an instant and consistent mesh reconstruction (*ISOMER*) algorithm to reconstruct high-quality 3D meshes from the multiple RGB images and normal maps, which fully integrates the color and geometric priors into mesh results. Both diffusion models are trained on a filtered version of the Objaverse dataset [7] with $\sim 50k$ 3D data. To enhance the quality and robustness, we design a series of strategies into our framework, including the noise offset channel in the multi-view diffusion training process to correct the discrepancy between training and inference [21], a stricter dataset filtering policy, and an expansion regularization to avoid normal collapse in mesh reconstruction. Overall, our method can generate high-fidelity, diverse, and multi-view consistent meshes from single-view wild images within 30 seconds, as shown in Figure 1.

We conduct extensive experiments on various wild 2D images with different styles. The experiments verify the efficacy of our framework and show that our Unique3D outperforms existing methods for high fidelity, geometric details, high resolution, and strong generalizability.

In summary, our contributions are:

- We propose a novel image-to-3D framework called Unique3D that holistically archives a leading level of high-fidelity, efficiency, and generalizability among current methods.
- We introduce a multi-level upscale strategy to progressively generate higher-resolution RGB images with the corresponding normal maps.
- We design a novel instant and consistent mesh reconstruction algorithm (*ISOMER*) to reconstruct 3D meshes with intricate geometric details and texture from RGB images and normal maps.

- Extensive experiments on image-to-3D tasks demonstrate the efficacy and generation fidelity of our method, unlocking new possibilities for real-world deployment in the field of 3D generative AI.

2 Related Work

Mesh Reconstruction. Despite the significant advancements in various 3D representations (*e.g.,* , SDF [61, 34], NeRF [31, 32], 3D Gaussian [16]), meshes remain the most widely used 3D format in popular 3D engines (*e.g.,* , Blender, Maya) with a mature rendering pipeline. Reconstructing high-quality 3D meshes efficiently from multi-view or single-view images is a daunting task in graphics and 3D computer vision. Early approaches usually adopt a laborious and complex photogrammetry pipeline with multiple stages, with techniques like Structure from motion (SfM) [1, 42, 51], Multi-View Stereo (MVS) [9, 43], and mesh surface extraction [35, 30]. Powered by deep learning and powerful GPUs, recent works [44, 45, 14, 67, 13, 60, 64] have been proposed to pursue higher efficiency and quality with gradient-based mesh optimization or even training a large feed-forward reconstruction network. However, their pipeline still suffers from heavy computational costs and struggles to adapt to complex geometry. To balance efficiency and quality, we propose a novel instant and high-quality mesh reconstruction algorithm in this paper that can reconstruct complex 3D meshes with intricate geometric details from sparse views.

Score Distillation for 3D Generation. Recently, data-driven large-scale 2D diffusion models have achieved notable success in image and video generation [39, 41, 66, 50]. However, transferring it to 3D generation is non-trivial due to curating large-scale 3D datasets. Pioneering works DreamFusion [36] proposes Score Distillation Sampling (SDS) (also known as Score Jacobian Chaining [55]) to distill 3D geometry and appearance from pretrained 2D diffusion models when rendered from different viewpoints. The following works continue to enhance various aspects such as fidelity, prompt alignment, consistency, and further applications [20, 37, 59, 5, 19, 54, 69]. However, such optimization-based 2D lifting methods are limited by long per-case optimization time and multi-face problem [48] due to lack of explicit 3D prior. As Zero123 [27] proves that Stable Diffusion [39] can be finetuned to generate novel views by conditioning on relative camera poses, one-2-3-45 [26] directly produce plausible 3D shapes from generated images in Zero123. Though it achieves high efficiency, the generated results show poor quality with a lack of texture details and 3D consistency.

Multi-view Diffusion Models for 3D Generation. To achieve efficient and 3D consistent results, some works [29, 28, 47, 57, 48] fine-tune the 2D diffusion models with large-scale 3D data [7] to generate multi-view consistent images and then create 3D contents using sparse view reconstruction. For example, SyncDreamer [28] leverages attention layers to produce consistent multi-view color images and then use NeuS [56] for reconstruction. Wonder3D [29] explicitly encodes the geometric information into 3D results and improves quality by cross-domain diffusion. Although these methods generate reasonable results, they are still limited by local inconsistency from multi-views generated by out-domain input images and limited generated resolution from the architecture design, producing coarse results without high-resolution textures and geometries. In contrast, our method can generate higher-quality textured 3D meshes with more complex geometric details within just 30 seconds.

3 Method

In this section, we introduce our framework, *i.e.,* Unique3D, for high-fidelity, efficient, and generalizable 3D mesh generation from a single in-the-wild image. Given an input image, we first generate four orthographic multi-view images with their corresponding normal maps from a multi-view diffusion model and a normal diffusion model. Then, we lift them to high-resolution space progressively, (Sec 3.1). Given high-resolution multi-view RGB images and normal maps, we finally reconstruct high-quality 3D meshes with our instant and consistent mesh reconstruction algorithm *ISOMER*, (Sec 3.2). ISOMER directly handles the case where the global normal of the same vertex is inconsistent across viewpoints to enhance the consistency. An overview of our framework is depicted in Figure 2.

3.1 High-resolution Multi-view Generation

We first explain the design of our high-resolution multi-view generation model that generates four orthographic view images from a single input image. Instead of directly training a high-resolution (2K)

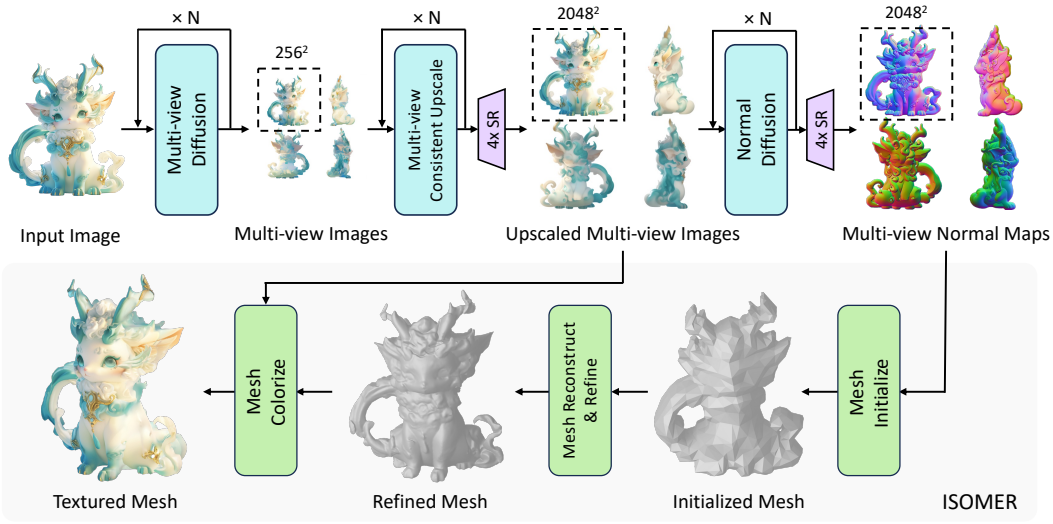

Figure 2: **Pipeline of our Unique3D**. Given a single in-the-wild image as input, we first generate four orthographic multi-view images from a multi-view diffusion model. Then, we progressively improve the resolution of generated multi-views through a multi-level upscale process. Given generated color images, we train a normal diffusion model to generate normal maps corresponding to multi-view images and utilize a similar strategy to lift it to high-resolution space. Finally, we reconstruct high-quality 3D meshes from high-resolution color images and normal maps with our instant and consistent mesh reconstruction algorithm *ISOMER*.

multi-view diffusion that would consume excessive computational resources, we adopt a multi-level generation strategy to upscale the generated resolution progressively.

High-resolution Multi-view Image Generation. Instead of training from scratch, we start with the initialization of the pre-trained 2D diffusion model using the checkpoint of Stable Diffusion [40] and encode multi-view dependencies to fine-tune it to obtain a multi-view diffusion model that is able to generate four orthographic view images (256 resolution) from a single in-the-wild image. It is worth noting that the images generated in this step have relatively low resolution and suffer from multi-view inconsistency in out-of-the-domain data. This significantly limits the quality of recent works [64, 60, 63, 29, 47]. In contrast, we address the multi-view consistency issue during the reconstruction phase (Sec 3.2). Given the generated four orthographic view images, we then finetune a multi-view aware ControlNet [70] to improve the resolution of images. This model leverages the four collocated RGB images as control information to generate corresponding clearer and more precise multi-view results. It enhances the details and ameliorates unclear regions, leading the resolution of images from 256 to 512. Finally, we employ a single-view super-resolution model [58] to further upscale the image by a factor of four, achieving a resolution of 2048 that offers sharper edges and details without disrupting the multi-view consistency.

High-resolution Normal Map Prediction. Using pure RGB images makes it extremely hard to reconstruct correct geometry. To effectively capture the rich surface details of the target 3D shape, we finetune a normal diffusion model to predict normal maps corresponding to multi-view color images. Similar to the above high-resolution image generation stage, we also employ the super-resolution model [58] to quadruple the normal resolution, which enables our method to recover high-fidelity geometric details, especially the accuracy of the edges.

To enhance the capability of the image generation model and the standard normal prediction model in producing high-quality images with uniform backgrounds, we adopt a channel-wise noise offset strategy [22]. This can alleviate the problem caused by the discrepancy between the initial Gaussian noise during sampling and the noisiest training sample.

3.2 ISOMER: An Efficient Method for Direct Mesh Reconstruction

Despite impressive results generated by recent popular image-to-3D methods [29, 23, 63, 13, 60] that follow the field-based reconstruction [44, 45, 46], they have limited potential for higher-resolution

applications as their computational load is proportional to the cube of the spatial resolution. *In contrast, we design a novel reconstruction algorithm directly based on mesh, where the computational load scales with only the square of the spatial resolution and relates to the number of faces, thus achieving a fundamental improvement.* This enables our model to efficiently reconstruct meshes with tens of millions of faces within seconds.

We now move to introduce our instant and consistent mesh reconstruction algorithm (*ISOMER*), which is a robust, accurate, and efficient approach for direct mesh reconstruction from high-resolution multi-view images. Specifically, the *ISOMER* consists of three main steps: (a) estimating the rough topological structure of the 3D object and generating an initial mesh directly; (b) employing a coarse-to-fine strategy to further approximate the target shape; (c) explicitly addressing inconsistency across multiple views to reconstruct high-fidelity and intricate details. Notably, the entire mesh reconstruction process takes no more than 10 seconds.

Initial Mesh Estimation. Unlike popular reconstruction methods based on signed distance fields [71] or occupancy fields [31], mesh-based reconstruction methods [10, 62] struggle with changing topological connectivity during optimization, which requires correct topological construction during initialization. Although initial mesh estimation can be obtained by existing methods like DMTet [44], they cannot accurately reconstruct precise details (*e.g.,* , small holes or gaps). To address the problem, we utilize front and back views to directly estimate the initial mesh, which is fast for accurate recovery of all topologically connected components visible from the front. Specifically, we integrate the normal map from the frontal view to obtain a depth map by

$$d(i,j) = \sum_{t} = 0^{i} n_x(t,j) \tag{1}$$

where $n_x(t,j)$ is the normal vector of the t-th pixel in the j-th row. Although the diffusion process generates pseudo normal maps, these maps do not yield a real normal field which is irrotational. To address this, we introduce a random rotation to the normal map before integration. The process is repeated several times, and the mean value of these integrations is then utilized to calculate the depth, providing a reliable estimation. Subsequently, we map each pixel to its respective spatial location using the estimated depth, creating mesh models from both the front and back views of the object. The two models are seamlessly joined through Poisson reconstruction, which guarantees a smooth connection between them. Finally, we simplify them into 2000 fewer faces for our mesh initialization.

Coarse-to-Fine Mesh Optimization. Building upon the research in inverse rendering [2, 33, 18], we iteratively optimize the mesh model to minimize a loss function. During each optimization step, the mesh undergoes differentiable rendering to compute the loss and gradients, followed by vertex movement according to the gradients. Finally, the mesh is corrected after iteration through edge collapse, edge split, and edge flip to maintain a uniform face distribution and reasonable edge lengths. After several hundred coarse-to-fine iterations, the model converges to a rough approximation of the target object's shape. The loss function for this part includes a mask-based loss

$$\mathcal{L}_{mask} = \sum_{i} \left\| \hat{M}_i - M_i^{pred} \right\|_2^2, \tag{2}$$

where $\hat{M}_i$ is the rendered mask under view i and M_i^{pred} is the predicted mask from previous subsection under view i. The mask-based loss regulates the mesh contour. Additionally, it includes a normal-based loss

$$\mathcal{L}_{normal} = \sum_{i} M_i^{pred} \otimes \left\| \hat{N}_i - N_i^{pred} \right\|_2^2, \tag{3}$$

concerning the rendered normal map $\hat{N}_i$ of the object and the predicted normal map N_i^{pred}, optimizing the normal direction in the visible areas, where $\otimes$ denotes element-wise production. We compute the final loss function as:

$$\mathcal{L}_{recon} = \mathcal{L}_{mask} + \mathcal{L}_{normal}. \tag{4}$$

To address potential surface collapse issues under limited-view normal supervision as shown in Figure 5-(b), we employ a regularization method called Expansion. At each step, vertices are moved a small distance in the direction of their normals, akin to weight decay.

ExplicitTarget Optimization for Multi-view Inconsistency and Geometric Refinement. Due to inherent inconsistencies in generated multi-view images from out-of-distribution (OOD) in-the-wild

input, no solution can perfectly align with every viewpoint. After the above steps, we can only reconstruct a model that roughly matches the shape but lacks detail, falling short of our pursuit of high-quality mesh. Therefore, we cannot use the common method that minimizes differences in all views, which would lead to significant wave-pattern flaws, as shown in Figure 5-(a). To overcome this challenge, finding a more suitable optimization target becomes crucial. Under single-view supervision, although a complete model cannot be reconstructed, the mesh shape within the visible area of that view can meet the supervision requirements with highly detailed structures. Based on this, we propose a novel method that assigns a unique optimization target for each vertex to guide the optimization direction. In contrast to the conventional implicit use of multi-view images as optimization targets, we **explicitly** define the optimization target with better robustness. We call this explicit optimization target as *ExplicitTarget* and devise it as follows:

(*ExplicitTarget*). Let $Avg(V, W) = \frac{\sum_i V_i W_i}{W_i}$ represent the weighted average function, and $V_M(v, i) : (\mathbb{N}^+, \mathbb{N}^+) \to \{0, 1\}$ represent the visibility of vertex v in mesh M under view i. $Col_M(v, i)$ Indicate the color of vertex v in viewpoint i. We compute the ExplicitTarget ET of each vertex in mesh M as

$$ET_M(v) = \begin{cases} Avg(Col_M(v,i), V_M(v,i)W_M(v,i)^2) & \text{, if } \sum_i V_M(v, \rangle) > 0 \\ \mathbf{0} & \text{, otherwise,} \end{cases} \tag{5}$$

where$W_M(v, i) = -\cos(N_v^{(M)}, N_i^{(view)})$ is a weighting factor that$N_v^{(M)}$ is the vertex normal of v in mesh M, and $N_i^{(view)}$ is the view direction of view i.

In the function $ET_M(\mathcal{I}, \mathcal{I}_m)$, the predicted color of vertex v is computed as the weighted sum of supervised views, with weights determined by the square of cosine angles. This is because the projected area is directly proportional to the cosine value, and the prediction accuracy is also positively correlated with the cosine value. The object loss function for ExplicitTarget is defined as

$$\mathcal{L}_{ET} = \sum_i M_i^{pred} \otimes \left\| \hat{N}_i - N_i^{ET} \right\|_2^2, \tag{6}$$

where N_i^{ET} is the rendering result of mesh M with $\{ET_M(\mathcal{I}, N^{pred}, v) | v \in M\}$ under the i-th viewpoint. The final optimization loss function is

$$\mathcal{L}_{refine} = \mathcal{L}_{mask} + \mathcal{L}_{ET}. \tag{7}$$

Towards this end, we finish the introduction of the *ISOMER* reconstruction process, which includes three stages: Initialization, Reconstruction, and Refinement.

Upon generating precise geometric structures, it is necessary to colorize them based on multi-view images. Given the inconsistencies across multi-view images, the colorizing process adopts the same method used in the refinement stage. Specifically, the colors of mesh M is $\{ET_M(\mathcal{I}, \mathcal{I}_{rgb}^{pred}, v) | v \in M\}$. Moreover, certain regions of the model may remain unobservable from the multi-view perspective, necessitating the coloring of these invisible areas. To address this, we utilize an efficient smoothing coloring algorithm to complete the task. More detailed and specific algorithmic procedures can be found in the Appendix.

4 Experiments

4.1 Experimental Setting

Dataset: Utilizing a subset of the Objaverse dataset as delineated by LGM [53], we apply a rigorous filtration process to exclude scenes containing multiple objects, low-resolution imagery, and unidirectional faces, leading to a refined dataset of approximately $50k$ objects. To address surfaces without thickness, we render eight orthographic projections around each object horizontally. By examining the epipolar lines corresponding to each horizontal ray, we identify $13k$ instances of illegitimate data. For rendering, we employ random environment maps and lighting to augment the dataset, thereby enhancing the model's robustness. To ensure high-quality generation, all images are rendered at a resolution of 2048×2048 pixels.

Network Architecture: The initial level of image generation is initialized with the weight of the Stable Diffusion Image Variations Model [40], while the subsequent level employs an upscaled

Figure 3: **Qualitative Comparison**. Our approach provides superior geometry and texture.

version fine-tuned from ControlNet-Tile [70]. The final stage uses the pre-trained Real-ESRGAN model [58]. Similarly, the initial stage of normal map prediction is initialized from the aforementioned Stable Diffusion Image Variations. Details of these networks are provided in the Appendix.

Reconstruction Details: The preliminary mesh structure is inferred from a normal map with a resolution of 256×256, which is then simplified to a mesh comprising $2,000$ faces. The reconstruction process involves 300 iterations using the SGD optimizer [3], with a learning rate of 0.3. The weight of expansion regularization is set to 0.1. Subsequent refinement takes 100 iterations, maintaining the same optimization parameters.

Training Details: The entire training takes around 4 days on 8 NVIDIA RTX4090 GPUs. The primary level of multiview image generation uses $30k$ training iterations with a batch size of $1,024$. The training of multi-view image upscaling involves $10k$ iterations with a batch size of 128. Normal

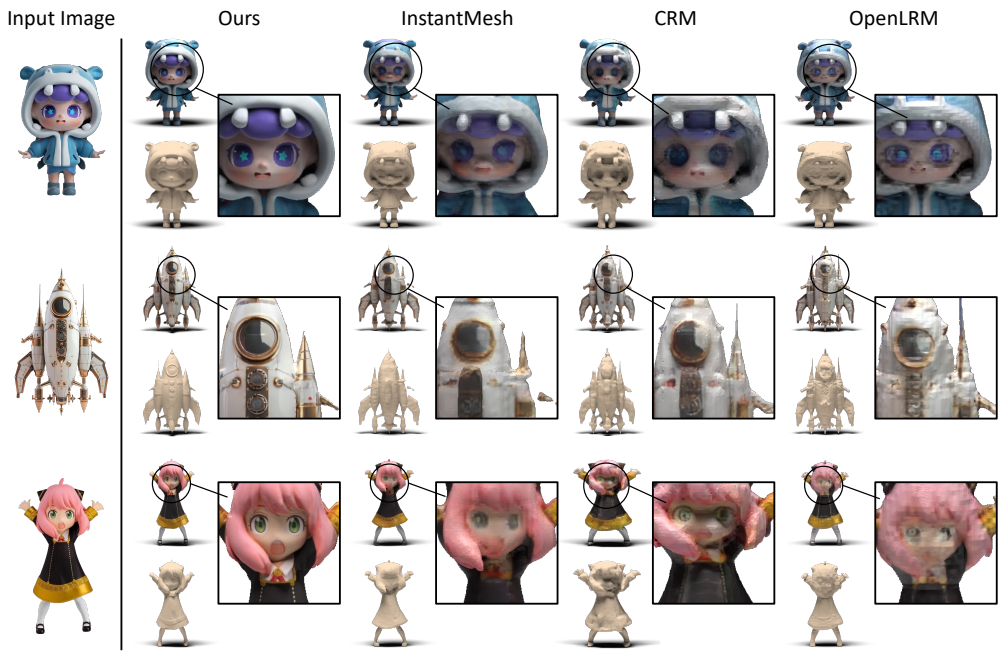

Figure 4: **Detailed Comparison**. We compare our model with InstantMesh [63], CRM [60] and OpenLRM [11]. Our models generates accurate geometry and detailed texture.

Table 1: Quantitative comparison results for mesh visual and geometry quality. We report the metrics of PSNR, SSIM, LPIPS and Clip-Similarity [38], ChamferDistance (CD), Volume IoU and F-score on GSO [8] dataset.

Method	**PSNR↑**	**SSIM↑**	**LPIPS↓**	**Clip-Sim↑**	**CD↓**	**Vol. IoU↑**	**F-Score↑**
One-2-3-45	16.1058	0.8874	0.1812	0.7782	0.0313	0.4142	0.5518
OpenLRM	18.0433	0.8957	0.1560	0.8416	0.0336	0.3947	0.5354
Wonder3D	18.0932	0.8995	0.1536	0.8535	0.0261	0.4663	0.6016
InstantMesh	<u>18.8262</u>	<u>0.9111</u>	<u>0.1283</u>	**0.8795**	0.0161	0.5083	0.6491
CRM	18.4407	0.9088	0.1366	0.8639	**0.0141**	<u>0.5218</u>	<u>0.6574</u>
Unique3D	**20.0611**	**0.9222**	**0.1070**	<u>0.8787</u>	<u>0.0143</u>	**0.5416**	**0.6696**
Unique3D w/o ET	20.0383	0.9199	0.1129	0.8675	0.0158	0.5320	0.6594
Wonder3D+ISOMER	18.6131	0.9026	0.1470	0.8621	0.0244	0.4743	0.6088

map prediction is trained for $10k$ iterations at a batch size of 128. Additional training specifics are accessible in the Appendix.

4.2 Comparisons

Qualitative Comparison: To highlight the advantages of our methodology, we perform a comprehensive comparison with existing works, including CRM [60], one-2-3-45 [26], Wonder3D [29], OpenLRM [11], and InstantMesh [63]. For a fair quality comparison, we choose to present samples previously selected in the referenced papers, originating from Wonder3D [29], SyncDreamer [28], CRM [60], and InstantMesh [63]. The results are shown in Figure 3. Our results clearly surpass the existing works in both geometric and material quality, thereby emphasizing the benefits of our approach in achieving high resolution and intricate details in both geometry and material. In addition to the above overall quality comparison, we further show the comparison of the details in Figure 4, highlighting the advantage of our method in high resolution. The reconstruction process of ISOMER is completed in under 10 seconds, while the entire procedure from the input image to high-precision mesh is accomplished in less than 30 seconds on an RTX4090.

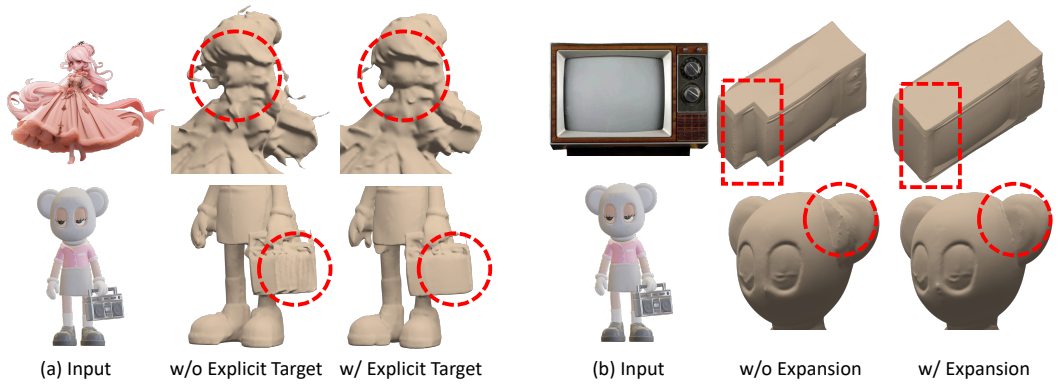

Figure 5: **Ablation Study on ISOMER**. (a) Without ExplicitTarget, the output mesh result has obvious defects. (b) Without expansion regularization, the output result collapses in some cases.

Figure 6: **Ablation on Colorize**. We show a comparison of whether or not to apply ExplicitTarget in coloring, and we can see that the group that does not use ExplicitTarget has significant artifacts, as there is no precise consistency across multiple views.

Quantitative Comparison: In line with previous work, we evaluate our results using the Google Scanned Objects (GSO) [8] dataset. We render frontal views at a resolution of 1024×1024 with Blender EEVEE as input for all methods. All generated mesh results are normalized to the bounding box $[-0.5, 0.5]$ to ensure alignment. The geometric quality is assessed by calculating the distance to the ground truth mesh using metrics such as Chamfer Distance (CD), Volume IoU, and F-Score. Concurrently, we render 24 views around the object, selecting one of $[0, 15, 30]$ for elevation angles and 8 evenly distributed azimuth angles spanning a full 360-degree rotation. We employ PSNR, SSIM, LPIPS, and Clip-Similarity [38] to evaluate the visual quality. The results are presented in Table 1. As evidenced in table, both our geometric and material quality outperform those of existing methods. We find that ISOMER can even be used to improve the consistency of other methods. For example, in Table 1, we replace Wonder3D's reconstruction method with ISOMER, which is not only faster but also of higher quality.

4.3 Ablation Study and Disscussion

We analyze the importance of ExplicitTarget and expansion regularization in ISOMER. We compare samples with and without ExplicitTarget and Expansion Regularization in figure 5. We clearly show the improvement of ExplicitTarget for geometry and the necessity of expansion regularization for reconstruction. ExplicitTarget notably improves reconstruction results in challenging cases, while expansion regularization avoids some possible collapses.

Mirroring the approach used for geometry, we will include additional experimental results in Figure 6 that illustrate the impact of the Explicit Target method on texture quality. Without the Explicit Target, the results are obviously flawed.

Table 2: Quantitative comparison results for ablation on 100 random samples with random rotation on GSO dataset.

Method	PSNR↑	SSIM↑	LPIPS↓	Clip-Sim↑	CD↓	Vol. IoU↑	F-Score↑
Unique3D	19.6744	0.9217	0.1101	0.8864	0.0118	0.5463	0.6833

Figure 7: **Ablation on Resolution**. The visualization of the generated multi-views images at different stages is shown. Multi-level super-resolution does not change the general structure, but only improves the detail resolution, allowing the model to remain well-detailed.

Additionally, we added a new test with randomly rotated objects sampled from $azimuth \in U[-180, 180], elevation \in U[-30, 30]$ in Table 2 to test robustness in non-front-facing views. The test results show that Unique3D still performs well in this case, and even the geometry prediction is more accurate.

We expand our study to include a qualitative comparison across various resolutions in order to demonstrate the differences between different resolutions in Figure 7. The results demonstrate the necessity of high resolution maps in generating high resolution meshes.

Challenging Examples: The majority of existing common samples are overly simplistic to effectively demonstrate the advantages of our study. Consequently, we select two complex samples: an object featuring detailed text and a photograph of a human, as shown in Figure 8. It is observable that our method exhibits exceptional mesh materials and geometry, even capable of sculpting geometric structures with textual detail. In the context of photographs, our reconstruction results are nearly on par with specialized image-to-character mesh generation methods.

Figure 8: **Challenging examples.**

5 Conclusion

In this paper, we introduce Unique3D, a pioneering image-to-3D framework that efficiently generates high-quality 3D meshes from single-view images with unprecedented fidelity and consistency. By integrating advanced diffusion models and the powerful reconstruction method ISOMER, Unique3D generates detailed and textured meshes within 30 seconds, significantly advancing the state-of-the-art in 3D content creation from single images.

Limitation and Future Works. Our method, while capable of generating high-fidelity textured meshes rapidly, faces challenges. The multi-view prediction model may produce less satisfactory predictions for skewed or non-perspective inputs. Furthermore, the geometric coloring algorithm currently does not support texture maps. In the future, we aim to enhance the robustness of the multi-view prediction model by training on a more extensive and diverse dataset.

References

[1] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 54(10):105–112, 2011.

[2] Mario Botsch and Leif Kobbelt. A remeshing approach to multiresolution modeling. In Jean-Daniel Boissonnat and Pierre Alliez, editors, *Second Eurographics Symposium on Geometry Processing, Nice, France, July 8-10, 2004*, volume 71 of *ACM International Conference Proceeding Series*, pages 185–192. Eurographics Association, 2004.

[3] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In Yves Lechevallier and Gilbert Saporta, editors, *19th International Conference on Computational Statistics, COMPSTAT 2010, Paris, France, August 22-27, 2010 - Keynote, Invited and Contributed Papers*, pages 177–186. Physica-Verlag, 2010.

[4] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[5] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. *CoRR*, abs/2303.13873, 2023.

[6] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems*, 36, 2024.

[7] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 13142–13153. IEEE, 2023.

[8] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas Barlow McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation, ICRA 2022, Philadelphia, PA, USA, May 23-27, 2022*, pages 2553–2560. IEEE, 2022.

[9] Yasutaka Furukawa, Carlos Hernández, et al. Multi-view stereo: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 9(1-2):1–148, 2015.

[10] Shubham Goel, Georgia Gkioxari, and Jitendra Malik. Differentiable stereopsis: Meshes from multiple views using differentiable rendering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 8625–8634. IEEE, 2022.

[11] Zexin He and Tengfei Wang. Openlrm: Open-source large reconstruction models. `https://github.com/3DTopia/OpenLRM`, 2023.

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[13] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. LRM: large reconstruction model for single image to 3d. *CoRR*, abs/2311.04400, 2023.

[14] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2821–2830, 2018.

[15] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*, 2023.

[16] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139:1–139:14, 2023.

[17] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[18] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular primitives for high-performance differentiable rendering. *ACM Trans. Graph.*, 39(6):194:1–194:14, 2020.

[19] Weiyu Li, Rui Chen, Xuelin Chen, and Ping Tan. Sweetdreamer: Aligning geometric priors in 2d diffusion for consistent text-to-3d. *arXiv preprint arXiv:2310.02596*, 2023.

[20] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. *CoRR*, abs/2211.10440, 2022.

[21] Shanchuan Lin, Bingchen Liu, Jiashi Li, and Xiao Yang. Common diffusion noise schedules and sample steps are flawed. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5404–5411, 2024.

[22] Shanchuan Lin, Bingchen Liu, Jiashi Li, and Xiao Yang. Common diffusion noise schedules and sample steps are flawed. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2024, Waikoloa, HI, USA, January 3-8, 2024*, pages 5392–5399. IEEE, 2024.

[23] Fangfu Liu, Hanyang Wang, Weiliang Chen, Haowen Sun, and Yueqi Duan. Make-your-3d: Fast and consistent subject-driven 3d content generation. *arXiv preprint arXiv:2403.09625*, 2024.

[24] Fangfu Liu, Diankun Wu, Yi Wei, Yongming Rao, and Yueqi Duan. Sherpa3d: Boosting high-fidelity text-to-3d generation via coarse 3d prior. *arXiv preprint arXiv:2312.06655*, 2023.

[25] Jian Liu, Xiaoshui Huang, Tianyu Huang, Lu Chen, Yuenan Hou, Shixiang Tang, Ziwei Liu, Wanli Ouyang, Wangmeng Zuo, Junjun Jiang, et al. A comprehensive survey on 3d content generation. *arXiv preprint arXiv:2402.01166*, 2024.

[26] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Advances in Neural Information Processing Systems*, 36, 2024.

[27] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 9264–9275. IEEE, 2023.

[28] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023.

[29] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. *arXiv preprint arXiv:2310.15008*, 2023.

[30] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*, pages 347–353. 1998.

[31] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: representing scenes as neural radiance fields for view synthesis. *Commun. ACM*, 65(1):99–106, 2022.

[32] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, 2022.

[33] Werner Palfinger. Continuous remeshing for inverse rendering. *Comput. Animat. Virtual Worlds*, 33(5), 2022.

[34] Jeong Joon Park, Peter Florence, Julian Straub, Richard A. Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 165–174. Computer Vision Foundation / IEEE, 2019.

[35] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. *ACM Trans. Graph.*, 22(3):313–318, 2003.

[36] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[37] Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skorokhodov, Peter Wonka, Sergey Tulyakov, et al. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *arXiv preprint arXiv:2306.17843*, 2023.

[38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021.

[39] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[40] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 10674–10685. IEEE, 2022.

[41] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022.

[42] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016.

[43] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pages 501–518. Springer, 2016.

[44] Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 6087–6101, 2021.

[45] Tianchang Shen, Jacob Munkberg, Jon Hasselgren, Kangxue Yin, Zian Wang, Wenzheng Chen, Zan Gojcic, Sanja Fidler, Nicholas Sharp, and Jun Gao. Flexible isosurface extraction for gradient-based mesh optimization. *ACM Trans. Graph.*, 42(4):37:1–37:16, 2023.

[46] Tianchang Shen, Jacob Munkberg, Jon Hasselgren, Kangxue Yin, Zian Wang, Wenzheng Chen, Zan Gojcic, Sanja Fidler, Nicholas Sharp, and Jun Gao. Flexible isosurface extraction for gradient-based mesh optimization. *ACM Trans. Graph.*, 42(4):37:1–37:16, 2023.

[47] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model. *CoRR*, abs/2310.15110, 2023.

[48] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *CoRR*, abs/2308.16512, 2023.

[49] Zifan Shi, Sida Peng, Yinghao Xu, Yiyi Liao, and Yujun Shen. Deep generative models on 3d representations: A survey. *CoRR*, abs/2210.15663, 2022.

[50] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

[51] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006.

[52] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[53] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. LGM: large multi-view gaussian model for high-resolution 3d content creation. *CoRR*, abs/2402.05054, 2024.

[54] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023.

[55] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A. Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. *CoRR*, abs/2212.00774, 2022.

[56] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021.

[57] Peng Wang and Yichun Shi. Imagedream: Image-prompt multi-view diffusion for 3d generation. *CoRR*, abs/2312.02201, 2023.

[58] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, Montreal, BC, Canada, October 11-17, 2021*, pages 1905–1914. IEEE, 2021.

[59] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *CoRR*, abs/2305.16213, 2023.

[60] Zhengyi Wang, Yikai Wang, Yifei Chen, Chendong Xiang, Shuo Chen, Dajiang Yu, Chongxuan Li, Hang Su, and Jun Zhu. CRM: single image to 3d textured mesh with convolutional reconstruction model. *CoRR*, abs/2403.05034, 2024.

[61] Wikipedia. Signed distance function — Wikipedia, the free encyclopedia. `http://en.wikipedia.org/w/index.php?title=Signed%20distance%20function&oldid=1189894340`, 2024. [Online; accessed 05-May-2024].

[62] Markus Worchel, Rodrigo Diaz, Weiwen Hu, Oliver Schreer, Ingo Feldmann, and Peter Eisert. Multi-view mesh reconstruction with neural deferred shading. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 6177–6187. IEEE, 2022.

[63] Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024.

[64] Yinghao Xu, Zifan Shi, Wang Yifan, Hansheng Chen, Ceyuan Yang, Sida Peng, Yujun Shen, and Gordon Wetzstein. GRM: large gaussian reconstruction model for efficient 3d reconstruction and generation. *CoRR*, abs/2403.14621, 2024.

[65] Fan Yang, Jianfeng Zhang, Yichun Shi, Bowen Chen, Chenxu Zhang, Huichao Zhang, Xiaofeng Yang, Jiashi Feng, and Guosheng Lin. Magic-boost: Boost 3d generation with mutli-view conditioned diffusion. *arXiv preprint arXiv:2404.06429*, 2024.

[66] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4):1–39, 2023.

[67] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pages 767–783, 2018.

[68] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *CoRR*, abs/2308.06721, 2023.

[69] Junliang Ye, Fangfu Liu, Qixiu Li, Zhengyi Wang, Yikai Wang, Xinzhou Wang, Yueqi Duan, and Jun Zhu. Dreamreward: Text-to-3d generation with human preference. *arXiv preprint arXiv:2403.14613*, 2024.

[70] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *CoRR*, abs/2302.05543, 2023.

[71] Fuqiang Zhao, Yuheng Jiang, Kaixin Yao, Jiakai Zhang, Liao Wang, Haizhao Dai, Yuhui Zhong, Yingliang Zhang, Minye Wu, Lan Xu, and Jingyi Yu. Human performance modeling and rendering via neural animated mesh. *ACM Trans. Graph.*, 41(6):235:1–235:17, 2022.

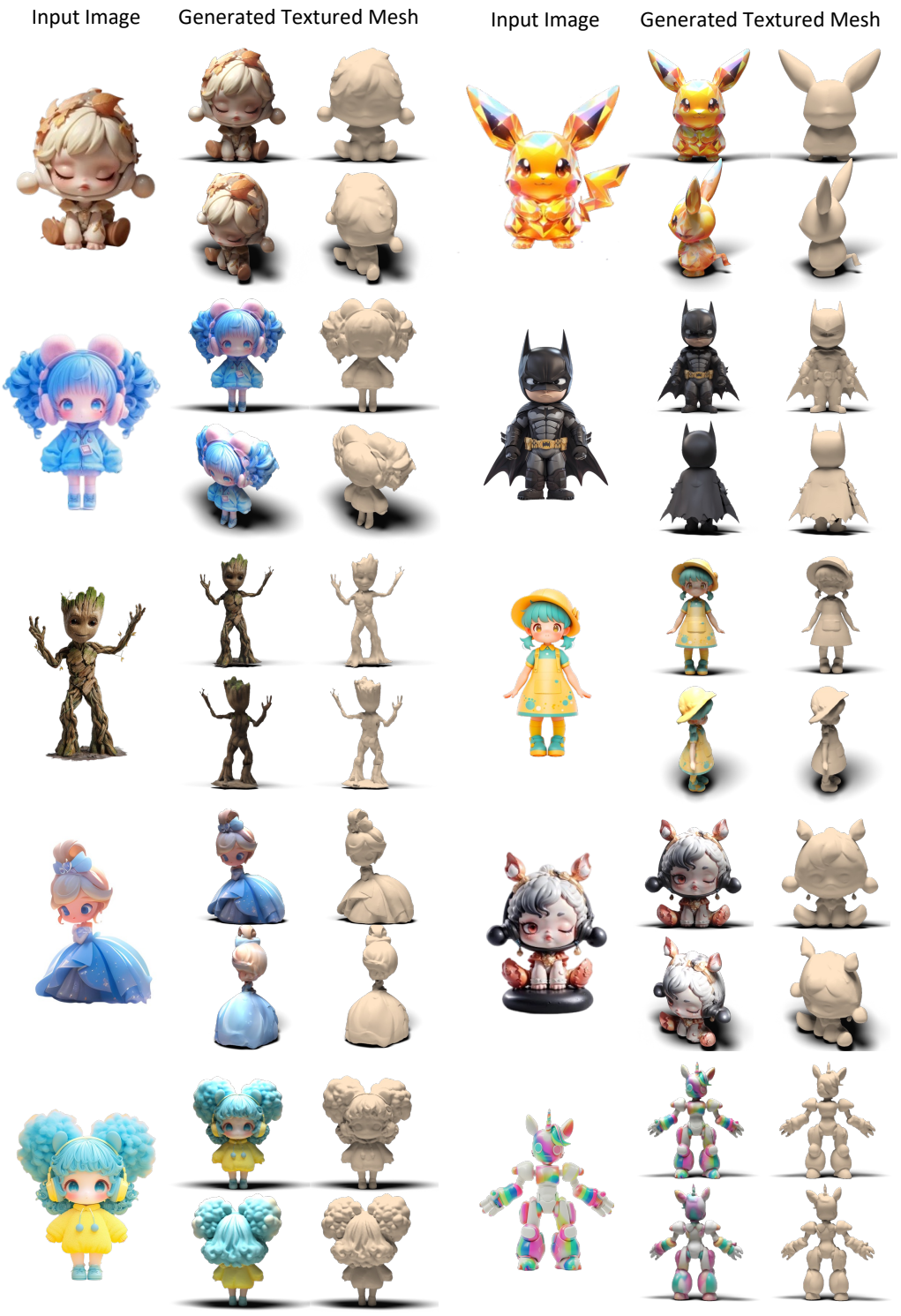

Figure 9: More generated results of our method from a single image.

A More Results

We provide more generation results of our method from a single image in Figure 9.

B Network Architecture and Training Details

Multi-view Image Generation: In this part, we develop a model based on the architecture of Stable Diffusion Image Variation [40] with two main modifications: (1). The use of a class embedding, which takes an integer from 0 to 3 as input, indicating the corresponding view indexes. (2). The simultaneous forward of four perspectives, where they are concatenated in the self-attention layers to achieve multi-view consistency.

For the training of the network, we utilize the following parameters:

- A learning rate of 10^{-4}.
- A batch size of 1024.
- A noise offset of 0.1.
- An SNR gamma of 5.0.
- An 8-bit Adam optimizer [17] with betas set to $(0.9, 0.999)$.
- An Adam weight decay of 0.01.
- An Adam epsilon of 10^{-8}.
- Gradient clipping with a norm of 1 to ensure training stability.

Multi-view Image Upscale: In this part, we aim to upscale merged low-resolution four-view images to a high resolution of 1024 pixels. To achieve this, we fine-tune the ControlNet-Tile network, leveraging StableDiffusion 1.5 as its backbone. Unlike traditional methods, we do not use text input; instead, we feed an empty text. Concurrently, we pass the input image through an IP-Adapter [68]. This approach allows the network to be guided in enhancing the multi-view details and achieving the desired resolution.

For the training of this network, we use the following parameters:

- A learning rate of 5×10^{-6}.
- A batch size of 128.
- A noise offset of 0.1.
- An SNR gamma of 5.0.
- An 8-bit Adam optimizer [17] with betas set to $(0.9, 0.999)$.
- An Adam weight decay of 0.01.
- An Adam epsilon of 10^{-8}.
- Gradient clipping with a norm of 1 to ensure training stability.
- Freeze parameters except for the ControlNet.

Normal Prediction Diffusion: In this part, we train a diffusion model that takes an RGB image as input and produces its corresponding normal map as output. This model is based on the Stable Diffusion Image Variation [40], with one key modification: a reference U-Net has been incorporated, which has an identical network structure and initialization as the original network. This reference U-Net provides pixel-wise reference attention to the main network exclusively at a new attention layer that added to the self-attention.

For the training of the network, we utilize the following parameters:

- A learning rate of 10^{-4} for main network.
- A learning rate of 10^{-5} for reference network.
- A batch size of 128.
- A noise offset of 0.1.
- An SNR gamma of 5.0.
- An 8-bit Adam optimizer [17] with betas set to $(0.9, 0.999)$.
- An Adam weight decay of 0.01.

- An Adam epsilon of 10^{-8}.
- Gradient clipping with a norm of 1 to ensure training stability.
- Freeze parameters except for the self-attention in reference attention.
- Train all parameters in the main network.

C Efficient Invisible Region Color Completion Algorithm

In our approach to mesh coloring using multiple viewpoints, we encounter a minor yet noteworthy challenge: the need to color regions that are not directly visible. Although these regions are typically sparse and inconspicuous. In field-based representations such as Signed Distance Fields [61], they often are the color of neighboring visible areas upon completion of the field optimization. To address this, we opt for a straightforward yet efficient algorithm that seamlessly spreads the colors of nearby visible regions into the invisible ones.

Our methodology employs a straightforward, multi-step color propagation algorithm, which stands out for its simplicity, swift execution, and reliability in delivering a reasonably detailed and nuanced color complement. This approach outperforms more complex, resource-intensive, and less stable techniques like using pre-trained inpainting diffusion models. The algorithm leverages the surrounding colors to gently fill in the invisible regions, with the detailed process outlined in Algorithm 1.

A critical aspect of the algorithm to consider is the potential for a stark color demarcation line if the process is halted immediately after all nodes have been colored. For example, in a one-dimensional scenario, if red is on the left and blue is on the right, separated by an uncolored section, stopping immediately will result in a high-contrast boundary. To mitigate this, we extend the color propagation process through a number of iterations to ensure a smooth color gradient. This allows the colors to gradually permeate throughout the entire connected component of the mesh that requires coloring, thus achieving a harmonious and visually coherent result.

Algorithm 1 Color Completion Algorithm

Input: Mesh M, list of invisible vertices Inv, list of color of all vertices C
Output: The completed color list C

$cnt \leftarrow 0$
$stage2 \leftarrow False$
$colored \leftarrow \emptyset$
for all vertices v in M **do**
  if $v \notin Inv$ **then**
    Append $True$ to $visible\_vertices$
  else
    Append $False$ to $visible\_vertices$
  end if
end for
while $stage2 == False$ or $cnt > 0$ **do**
  for all i in Inv **do**
    $colored\_neighbors \leftarrow$ list of vertices directly connected to i in M that have $colored == True$
    if $colored\_neighbors! = \emptyset$ **then**
      $colored[i] \leftarrow True$
      $C[i] \leftarrow$ mean$(C[colored_neighbors])$
    else
      $colored[i] \leftarrow False$
    end if
  end for
  if all elements of $colored$ are $True$ **then**
    $stage2 \leftarrow True$
    $cnt \leftarrow cnt - 1$
  else
    $cnt \leftarrow cnt + 1$
  end if
end while
return C

Algorithm 2 ExplicitTarget Algorithm

Input: Multi-view image list $imgs$, initial mesh model M
Output: Model M' with vertex colors set to **ExplicitTarget**

$M' \leftarrow M$
Set the color of M' to the vertex normals of M'
for all vertices v in the vertex set of M' **do**
    $tot\_weight \leftarrow 0$
    $tot\_color \leftarrow \vec{0}$ ▷ Initialize to zero vector
    for all images im in $imgs$ **do**
        if vertex v is not visible in the viewpoint of im **then**
            continue
        end if
        $ci \leftarrow$ the color of vertex v in image im
        $wi \leftarrow$ the square of the cosine of the angle between the vertex normal of v and the view direction from im to v
        $tot\_weight \leftarrow tot\_weight + wi$
        $tot\_color \leftarrow tot\_color + wi \cdot \vec{ci}$
    end for
    if $tot\_weight > 0$ **then**
        Set the color of vertex v in M' to tot_color/tot_weight
    end if
end for
return M'

D ExplicitTarget algorithm

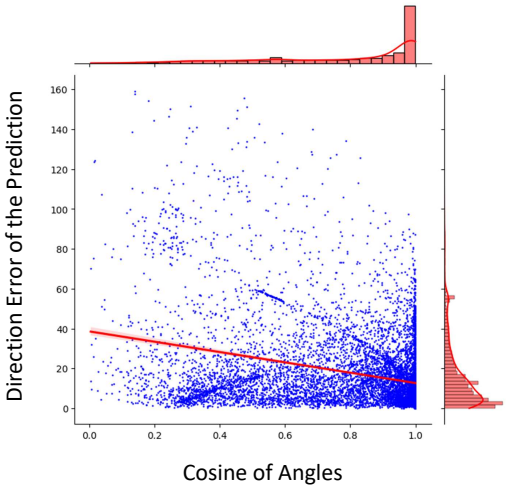

Figure 10: Correlation between prediction value and prediction errors.

In Algorithm 2, we demonstrate the detailed computation of ExplicitTarget. Specifically, we set an optimization target for each vertex, which is a weighted sum of the supervised signals from the visible views of the vertex. The weights are determined by two factors: the projected area of the nearby surface and the confidence level in the accuracy of the normals, which are used to calculate the weights. In Figure 10, we show the relationship between the normal results predicted by multiView diffusion and the accuracy of the predictions on the Objaverse [7] validation set. The results indicate that the closer the angle between the predicted normal and the vertical to the current viewpoint is, the lower the accuracy of the prediction. There is a negative correlation between these two factors, with a Pearson correlation coefficient of -0.304.

E More on Mesh Optimizations

Edge Collapse: This operation is used to avoid and heal defects in the mesh. It involves selecting an edge within a triangle and collapsing it to the other edge, effectively merging the two triangles into a single triangle. This process can help to eliminate narrow triangles that might be causing issues in the mesh, such as those that are too thin to accurately represent the surface they are approximating. Edge collapse can prevent the creation of topological artifacts and maintain the quality of the mesh. *Edge Split*: This is the opposite of edge collapse. In edge split, an edge that is longer than a specified maximum length is divided into two, creating new vertices at the midpoint of the edge. This operation is used to refine the mesh, ensuring that the local edge length is kept close to the optimal length. It helps to maintain the quality of the mesh by avoiding edges that are too long, which could lead to an inaccurate representation of the surface. *Edge Flip*: Edge flip is an operation that adjusts the connectivity of the mesh to improve its quality. It involves flipping an edge within a triangle to connect two non-adjacent vertices, effectively changing the triangulation of the mesh. This can help to maintain the degree of the vertices close to their optimal value, which is typically six for internal vertices (or four for boundary vertices). The goal of these operations is to improve the mesh quality while avoiding defects and ensuring that the mesh accurately represents the target geometry.

F Ablation Study on Mesh Initialization

Figure 11: **Ablations on Mesh Initialization**. We compare the results of using our fast initialization method, versus using a sphere as an initialization.

We compare the different mesh initialization methods and their results. One is our proposed fast initialization method, and the other is using spheres as initialization objects, a common practice in mesh-based reconstruction techniques. Figure F illustrates the problem of the mesh reconstruction method that fails to modify its topological structure. For example, in the first row, the model cannot achieve a hollow structure by direct optimization because their topologies are inherently different. However, as shown in the second row, even though the topologies are different, the optimization method can still provide approximate results. For instance, the sphere-based initialization can shape the handle on the right side, even though the handle is incomplete. The sphere-based initialization can sometimes produce even more accurate results than our proposed method, as seen in the third row. These experiments demonstrate that our method is robust to initialization. Aiming for a better ability to generalize, we chose to use our fast initialization.

G User Study

For user study, we render 360-degree videos of subject-driven 3D models and show each volunteer with five samples of rendered video from a random method. They can rate in four aspects: 3D consistency, subject fidelity, prompt fidelity, and overall quality on a scale of 1-10, with higher scores indicating better performance. We collect results from 30 volunteers shown in Table 3. We find our method is significantly preferred by users over these aspects.

Table 3: Quantitative comparison results on the multi-view consistency, subject fidelity (related to geometric and texture details), prompt fidelity (related to the alignment of input single image), and overall quality score in a user study, rated on a range of 1-10, with higher scores indicating better performance.

Method	Multi-view Consistency	Subject Fidelity	Prompt Fidelity	Overall Quality
One-2-3-45 [26]	5.46	4.78	6.93	5.79
OpenLRM [13]	6.72	7.16	6.92	7.15
SyncDreamer [28]	5.71	7.52	4.06	5.92
Wonder3D [29]	8.67	7.80	7.39	8.14
InstantMesh [63]	8.31	7.68	7.91	8.43
GRM [64]	6.93	7.42	6.02	7.38
CRM [60]	7.95	8.53	8.03	8.25
Ours	**9.26**	**8.74**	**8.52**	**9.02**

H Social Impact

Positive Impacts: The Unique3D framework can democratize 3D content creation, making it easier for artists and designers to produce 3D models from single images, which can lead to increased innovation and a surge in creative applications across various industries including gaming, film, and education.

Negative Impacts: On the flip side, the ease of generating high-quality 3D models raises concerns about potential misuse, such as creating deepfakes, and could lead to job displacement for traditional 3D modelers. Additionally, there may be challenges related to intellectual property and privacy if the technology is used irresponsibly.

I Licenses for Used Assets

Stable Diffusion [40] is under CreativeML Open RAIL M License.

Objaverse [7] is under ODC-By v1.0 license.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We prove our claim through quantitative and qualitative experiments.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Yes, in the Conclusion section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: There is no theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Yes, we released the inference code on GitHub.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Yes, the paper provides open access to the data and code, along with comprehensive instructions.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes, see the Experiments section and Appendix for details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: No, error bars are not available.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Yes, in the Experiments section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Yes, the research in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Yes, in Appendix.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The license of assets are listed in Appendix.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: No new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: The paper involves user study with human subjects. We invited 30 volunteers from the college to participate in our user study.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: In the study described in this paper, no potential risks were identified for participants. The research design was carefully crafted to ensure the safety and well-being of all individuals involved.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.