T^{3}Bench: Benchmarking Current Progress in Text-to-3D Generation

Yuze He[1*], Yushi Bai[1*], Matthieu Lin[1], Wang Zhao[1], Yubin Hu[1], Jenny Sheng[1], Ran Yi[2], Juanzi Li[1], and Yong-Jin Liu[1✉]

[1] Tsinghua University
[2] Shanghai Jiao Tong University
{hyz22,bys22}@mails.tsinghua.edu.cn, liuyongjin@tsinghua.edu.cn

Abstract. Recent methods in text-to-3D leverage powerful pretrained diffusion models to optimize NeRF. Notably, these methods are able to produce high-quality 3D scenes without training on 3D data. Due to the open-ended nature of the task, most studies evaluate their results with subjective case studies and user experiments, thereby presenting a challenge in quantitatively addressing the question: How has current progress in Text-to-3D gone so far? In this paper, we introduce T^{3}Bench, the first comprehensive text-to-3D benchmark containing diverse text prompts of three increasing complexity levels that are specially designed for 3D generation. To assess both the subjective quality and the text alignment, we propose two automatic metrics based on multi-view images produced by the 3D contents. The quality metric combines multi-view text-image scores and regional convolution to detect quality and view inconsistency. The alignment metric uses multi-view captioning and GPT-4 evaluation to measure text-3D consistency. Both metrics closely correlate with different dimensions of human judgments, providing a paradigm for efficiently evaluating text-to-3D models. The benchmarking results, shown in Fig. 1, reveal performance differences among an extensive 10 prevalent text-to-3D methods. Our analysis further highlights the common struggles for current methods on generating surroundings and multi-object scenes, as well as the bottleneck of leveraging 2D guidance for 3D generation. Our project page is available at: https://t3bench.com.

Keywords: Text-to-3D · Evaluation · Generative Models

1 Introduction

> *It is a narrow mind which cannot look at a subject from various points of view.*
> — *George Eliot*

Equipping machines with the ability to automatically generate 3D objects and scenes from text descriptions has long been an ambitious and ongoing pursuit. Recent methods, such as diffusion model [10,31] and NeRF [8,23,48], have significantly improved the effectiveness of text-to-3D methods, empowering potential applications ranging from arts realization to industrial design.

* Equal contribution

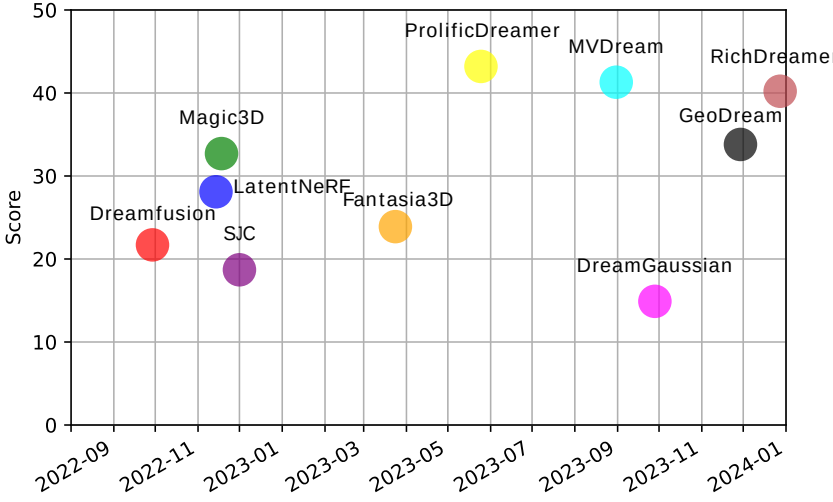


Fig. 1: The average scores of 10 prevalent text-to-3D methods on T^{3}Bench, computed by the mean of quality & alignment metrics.

However, there lacks a systematic approach to benchmarking current progress on text-to-3D methods, which is most prominently reflected in two aspects: (a) A lack of a standard set of diverse, challenging test textual inputs. (b) An absence of a set of automatic and comprehensive evaluation metrics to quantitatively measure the quality of the generated 3D scenes. Specifically, previous works [16, 36, 37] mostly adopt simple object or scene prompts for evaluation, and largely rely on subjective user experiments. Several works [24, 27, 42, 45] assess 3D generation quality by rendering the generated 3D model into a single 2D image and measuring its alignment with the text prompt through CLIP cosine distance or CLIP R-precision. Nevertheless, they only consider **one view** of the 3D scene, failing to assess the overall 3D quality.

To facilitate further research in this direction, we introduce T^{3}Bench, the first comprehensive text-to-3D benchmark. For a careful and thorough assessment, we build the benchmark to accurately reflect the primary challenges of current text-to-3D approaches. This includes their scalability and robustness in generating a variety of 3D scenes, the quality and view consistency of these generated scenes, and the correctness or alignment of these 3D scenes with their respective texts. Specifically, we devise three prompt suites incorporating diverse 3D scenes and with increasing complexity, including *Single object*, *Single object with surroundings*, and *Multiple objects*. We also propose two automatic evaluation metrics that both take **multi-view information** into consideration, focusing on assessing the subjective quality of the generated 3D scenes and its alignment with the textual prompt respectively. To calculate these two metrics, we first employ multi-focal and multi-view capturing to obtain a set of 2D images from the generated 3D scenes. The *quality* metric individually scores these multi-view

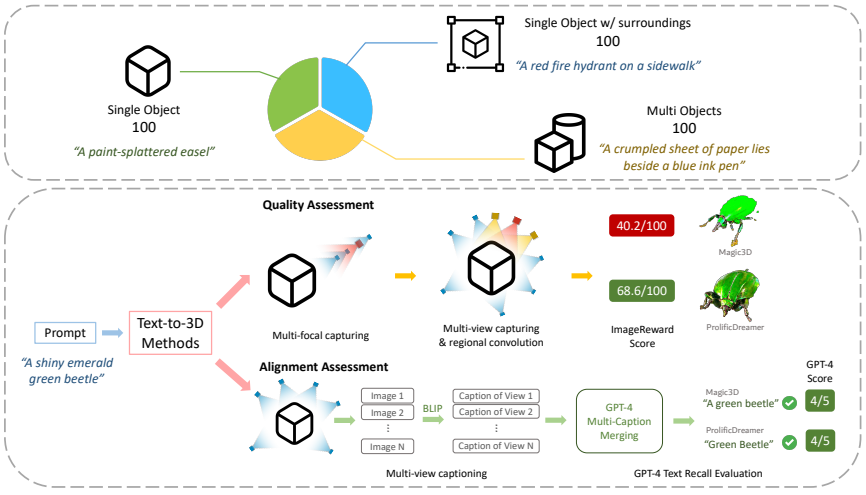


Fig. 2: The overview of our T^{3}Bench benchmark pipeline.

images with text-image scoring models (CLIP [30], ImageReward [46]), and then combines them into one overall quality measurement using *regional convolution*, which also effectively detects the infamous Janus problem (view inconsistency) in prevalent text-to-3D models [11, 27]. On the other hand, the *alignment* metric utilizes multi-view captioning and GPT-4 evaluation to measure how closely the 3D information aligns with the textual information in the input text prompt. Our user experiments show that both metrics correlate closely with human scorings in 1-5 scale (with a Spearman correlation higher than 0.75), demonstrating them as efficient and accurate measurements.

As the **first** attempt to benchmark current text-to-3D methods, T^{3}Bench yields fruitful results. Our benchmark reveals the strengths and weaknesses across 10 prevalent text-to-3D methods, as well as their common insufficiency when faced with more complicated 3D scenes, such as those involving multiple objects. We also analyze the correlation between the performance of text-to-3D methods and the quality of the 2D guidance generated by diffusion models, showing that the primary hurdle for text-to-3D mainly lies in the transition from 2D to a consistent 3D scene.

2 Related Works

Text-to-3D. Predominant works in text-to-3D [4, 16, 22, 26, 28, 44] circumvent the need for 3D training data by using large pretrained text-to-image diffusion models [32, 34]. However, these approaches suffer from inconsistency between views. Notably, the proposed score distillation loss [26] does not take into account the consistency between views as the diffusion model mimics a stochastic process [10]. On the one hand, ProlificDreamer [44] proposes a variational formulation of the score distillation loss to consider the stochasticity in the diffusion

process. On the other hand, some researchers propose to fine-tune the diffusion model to improve its consistency across views [50]. However, current metrics do not adequately consider the 3D nature of the generated results, which makes it difficult to compare the effectiveness of different methods. Prior work has relied either on labor-intensive user studies [44] or CLIP R-precision [30], which does not consider 3D consistency. While early attempts have been made to measure 3D consistency [11], these efforts only capture one aspect of the problem and overlook crucial metrics such as quality and prompt alignment.

Text-to-image Generation and Evaluation. With the development of diffusion models [10], text-based image generation has experienced significant progress in recent years [32, 34]. These models excel at complex tasks like editing and composition [3, 9]. However, comparing their capabilities in text-based generation is challenging due to the open-ended nature of the task [2]. Prior work in text-to-image generation introduced DrawBench [34], a comprehensive set of prompts aiming to evaluate various aspects, including color understanding, object recognition, and spatial relations. Other approaches leverage CLIP [30] and BLIP [15] to measure the similarity between text and generated images by using these models as scorers to gauge prompt alignment. In a similar vein, the Aesthetic score [35] employs the CLIP model to predict image aesthetics. While these methods assess alignment and quality to some extent, they fall short in considering multiple properties like toxicity, quality, and alignment. To encompass these diverse properties into a single model, ImageReward [46] proposes training a reward model via reinforcement learning from human feedback. Results show that this reward model better aligns with human preferences. Although evaluation for 3D generation can draw on text-to-image evaluation methods, it is important to note the major difference between the two: 3D contains semantic information from multiple viewpoints rather than a single view.

3 Method

This section presents the methodology in constructing T^{3}Bench, including the design and generation of text prompts, the unification of 3D representations, and the introduction of two novel evaluation metrics — the quality assessment and the alignment assessment.

3.1 Diverse Prompts with Increasing Complexity

While there are some widely used text-to-image prompt sets, such as DrawBench [33] and DALL-EVAL [5], many of the prompts in these benchmarks pose substantial challenges for existing text-to-3D methods and lack an adequate degree of distinction. Certain prompts, for instance, are excessively lengthy, while others incorporate complex aspects such as counting, leading to poor 3D scenes generated by all current text-to-3D methods. Therefore, a new set of prompts needs to be specifically crafted for evaluating prevalent text-to-3D methods.

We observe that current text-to-3D approaches demonstrate relatively robust performance on prompts with a single object. However, their performance notably declines on text prompts that include environmental surroundings or multiple objects. Such deficiency is partly due to the utilization of 2D supervision, which cannot ensure consistency amongst different viewpoints. With these observations, we design three prompt sets with increasing complexity to perform a targeted evaluation of text-to-3D approaches, namely *Single object*, *Single object with surroundings*, and *Multiple objects*. The *Single object* set represents the simplest scenario to establish a baseline level of performance, and the other two prompt sets introduce increased levels of difficulty by incorporating additional information, *i.e.,* surroundings or multiple objects.

To generate these prompt sets, we first use GPT-4 [25] to generate a large pool of candidate prompts, and then manually filter out prompts that contain proper nouns or toponyms. Subsequently, we utilize ROUGE-L [17] to quantify prompt similarity and gradually remove highly similar prompts until there remains a number of N distinct prompts with significant diversity in each prompt set.

3.2 Unified 3D Representation

Different text-to-3D methods employ various 3D representations during generation, such as NeRF [23] and 3D mesh. From a testing perspective, a 3D mesh is more conducive than NeRF due to its explicit geometric structure, which facilitates localization and normalization. Moreover, the primary use of text-to-3D is to obtain editable 3D assets that can be applied in fields such as virtual reality and gaming. Considering the purpose and practical applications, 3D mesh is a more suitable unified representation for benchmarking text-to-3D methods. We convert NeRF generated by text-to-3D methods into a 3D mesh using either DMTet [38] or Marching Cube [19], and choose the one that produces superior results. This makes subsequent evaluations more convenient while encourages the generation of 3D scenes with more compact and clear geometry.

3.3 Evaluation Metrics on Quality and Alignment

Overview The evaluation of text-to-3D methods remains challenging due to the need to fully account for the quality, view consistency, and text alignment of the generated 3D scenes.

Our evaluation metrics primarily focus on two dimensions [13] that typically reflect the effectiveness of text-to-3D methods: (1) the subjective quality of the generated 3D scene, and (2) the degree of alignment between the generated 3D scene and the input text prompt. To assess quality, we devise a scoring mechanism that comprises multi-focal and multi-view capturing, and utilizes text-image scoring models to obtain an overall quality measurement of the generated 3D scene. As for the textual alignment, we develop a scoring metric based on multi-view captioning and GPT-4 evaluation.

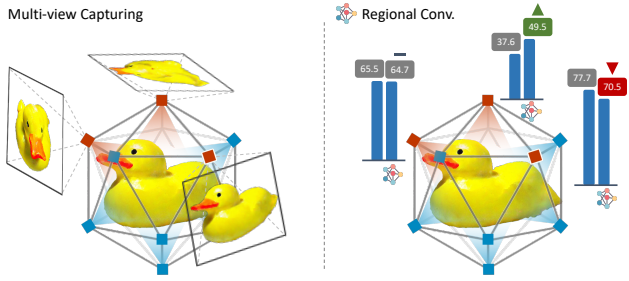


Fig. 3: Demonstration of scores at different viewpoints after multi-view capturing and regional convolution. Here, we use a level-0 icosahedron for a schematic illustration, please refer to Fig. 6 in the supplementary material for more details.

Quality Assessment Since the spatial geometry information is crucial for the generated 3D scenes, evaluation from a single view is incapable of assessing the quality of the generated results. We believe a comprehensive and reliable 3D quality assessment should take into account the following aspects: (a) *Viewpoint selection*: choosing an appropriate viewpoint can better reflect the quality of the 3D scene, particularly potential object occlusions; (b) *Area coverage*: it is essential to simultaneously examine the current viewpoint and adjacent areas. By doing so, the assessment can take into account a more global geometry, thereby avoiding a collapse to a local optimal view that leads to failure in detecting 3D consistency issues like the Janus problem.

To meet these requirements, we incorporate a delicate capturing and scoring procedure to evaluate the quality of the 3D generation. The following steps outline our method:

Mesh Normalization. We convert the generated 3D scene into a mesh and scale it proportionally in the x, y, and z directions, allowing the mesh to fit within a cube with a range of $[-1, 1]$ on all three dimensions. This helps to roughly determine the mesh's range for subsequent capturing.

Multi-Focal Capturing. Capturing a 2D image using a fixed focal length from a single location may yield inaccurate evaluation results. This is because the information in the captured image may be incomplete when the focal length is too long, and may occupy only a small portion in the frame when the focal length is too short. To address this issue, we employ five different focal lengths to capture the mesh at each location and select the best focal length based on the highest text-image score.

Multi-View Capturing. To capture the 3D scene as completely as possible, we construct an icosahedron with a radius of 2.2 around the origin and capture the 3D scene from all the vertices of the icosahedron (see an illustration of icosahedron in Fig. 3,6). As text-image scoring models may be sensitive to rotation, we ensure that the plane formed by the up vector and look-at vector during capture contains the vertical axis. In practice, we use a level-2 icosahedron and capture from 161 locations.

Scoring and Regional Convolution. We employ text-image scoring models, e.g., CLIP [30] and ImageReward [46], to score the 2D views from all 161 icosahedron vertices along with the textual prompt. To capture a more global feature, we apply a pooling operator to the score at each location. Standard averaging of scores across all locations may not be appropriate, as most views are not suitable for evaluation (e.g., top or bottom), and this approach may oversmooth the actual performance. Meanwhile, taking the overall maximum scores may overlook the view inconsistency issue. Therefore, we design a *regional convolution mechanism* to smooth out the score over each local region. We treat the icosahedron as a graph composed of vertices and edges, and perform mean pooling on the graph with the following recursive formula:

$$s_i^{(t+1)} = \frac{1}{w|N(i)|+1}\left(s_i^{(t)} + w\sum_{j\in N(i)} s_j^{(t)}\right), \tag{1}$$

where $s_i^{(t)}$ is the score of point i on the icosahedron at the t-th iteration, $N(i)$ is the set of neighboring points of i, and $|N(i)|$ is the number of neighbors of i. The superscript $(t+1)$ denotes the score after the $(t+1)$-th iteration. We choose a total of $t = 3$ iterations of mean pooling and convolution weight $w = 1$ as we empirically find that they ensure a balance between adequate smoothing and over-smoothing (please refer to Sec. 9 for more details).

After these steps, we select the highest score from all viewpoints as the final quality score for the 3D generation.

Alignment Assessment In addition to the evaluation from the quality aspect, the alignment between 3D semantic information and text is another crucial aspect that should be considered. To measure the alignment between different modalities, we first perform 3D-to-text captioning on the 3D scene and then compute the similarity between the caption and the textual prompt.

Directly utilizing image captioning methods such as BLIP [15] on a single view may fail to reflect the comprehensive information of a 3D object. To this end, we utilize a 3D-to-text captioning pipeline similar to Cap3D [20]. Initially, a level-0 icosahedron consisting of 12 vertices is established around the origin. This icosahedron captures the 3D scene on the 12 locations, each of which is captioned using BLIP. We then employ GPT-4 [25] to merge these captions (detailed in Sec. 7.2 of supplementary material), resulting in a 3D caption for the object.

Upon obtaining the 3D caption, we need to measure its alignment with the original prompt, particularly concerning the *recall* of the original prompt within the caption. Specifically, we observe that the text-to-3D methods might generate features not mentioned in the prompt (e.g., a red beak feature on a rubber duck), which may be reflected in the caption provided by BLIP. Such additional features should not be considered misalignments, even though many similarity-based scoring methods (BLEU, BERTScore [49]) might assign them lower scores. To assess the text recall, we adopt ROUGE-L [17]. We also incorporate GPT-4 as

text recall evaluators, drawing upon their demonstrated ability to mimic human experts in data annotation and evaluation [1]. Here is the prompt we use:

Prompt: You are an assessment expert responsible for prompt-prediction pairs. Your task is to score the prediction according to the following requirements:
1. Evaluate the recall, or how well the prediction covers the information in the prompt. If the prediction contains information that does not appear in the prompt, it should not be considered as bad.
2. If the prediction contains correct information about color or features in the prompt, you should also consider raising your score.
3. Assign a score between 1 and 5, with 5 being the highest. Do not provide a complete answer; give the score in the format: 3
Prompt: A photographer is capturing a beautiful butterfly with his camera
Prediction: A man photographing a butterfly near a tree and map, surrounded by plants
Answer: 4

4 Experiments

4.1 Metric Evaluation

In order to validate the reliability of our proposed metrics, we conduct a human-centered evaluation. Expert evaluators are tasked with manually assigning scores to 3D scenes generated by 7 different methods (detailed in Sec. 4.2) on 30% of all the prompts in T^{3}Bench. This results in a total of 1,260 scores. The human annotations span two dimensions: quality, which concerns the subjective quality of the generated results, and alignment, which focuses on the extent to which the generated content covers the original prompt. These evaluations are quantified using a 1-5 Likert scale. Subsequently, we measure the correlation between the proposed metrics and human annotations using Spearman's ρ, Kendall's τ, and Pearson's ρ correlation coefficients.

Table 1: Overview of evaluation metrics used in previous works.

	User Study	CLIP R-Precision	CLIP Similarity
Dream Fields [12]		✓	
DreamFusion [27]		✓	
Magic3D [16]	✓		
ProlificDreamer [44]	✓		
MVDream [40]	✓		
GaussianDreamer [47]			✓
Instant3D [14]		✓	

We list the commonly used metrics for text-to-3D evaluation in Table 1. The CLIP R-Precision and CLIP Similarity metrics used in previous works only

Table 2: The correlation of different combinations of evaluation methods with human annotations.

	CLIP Similarity	CLIP R-Precision	**Multi-view Quality** CLIP	**Multi-view Quality** ImageReward	**Multi-view Alignment** ROUGE-L	**Multi-view Alignment** GPT-4
Quality						
Spearman (ρ) ↑	0.638	0.464	0.749	**0.784**	0.407	0.593
Kendall (τ) ↑	0.496	0.420	0.597	**0.636**	0.310	0.508
Pearson (ρ) ↑	0.621	0.457	0.727	**0.780**	0.393	0.554
Alignment						
Spearman (ρ) ↑	0.638	0.479	0.745	0.722	0.567	**0.780**
Kendall (τ) ↑	0.495	0.433	0.590	0.572	0.442	**0.701**
Pearson (ρ) ↑	0.627	0.475	0.730	0.713	0.574	**0.774**

consider one view, and we compare them with our proposed multi-view based metrics. Specifically, for our quality metric, we consider CLIP [30] and ImageReward [46] as single-view scoring methods; for the alignment metric, we explore the use of ROUGE-L and GPT-4 to measure text recall. CLIP R-Precision uses CLIP to retrieve the correct caption among a set of distractors given a rendering of the content. Following prior works [12, 26], we use the 153 prompts from the object-centric COCO [18] validation subset of Dream Fields [12] as the negative prompt set.

We report the results in Tab. 2. Drawing conclusion from the first four columns, we validate that our proposed multi-view based metrics are superior to single-view examination. Moreover, compared to ROUGE-L, GPT-4 provides a more reliable assessment of alignment, as depicted by the last two columns. These findings justify the design of our processing and scoring methods in Sec. 3.3. The inherent characteristics of retrieval-based metrics, which provide assessments that are comparative rather than absolute, also result in misalignment with human perceptual processes. Overall, we observe that Multi-view capturing + ImageReward and 3D captioning + GPT-4 scoring align most closely with quality and alignment aspects as annotated by human experts, respectively. We thus employ these combinations as the default quality and alignment metrics in our benchmark, throughout the rest of the paper.

Janus problem analysis. The Janus problem, or multi-view inconsistency issue, arises when using Stable Diffusion for guidance, as it may not always generate accurate front, side, or back views for training. Consequently, this can lead to the regeneration of content described by the text prompt, and the most canonical view (e.g., the front piggy face) of an object appears in other views (e.g., the back view of a piggy). In the following, we validate that **3D scenes with the Janus problem can be reflected in our multi-view metrics**.

Intuitively, given a large number of viewpoints, for an object with the Janus problem, many views will show wrong results and lead to a decline in the quality score. After employing regional convolution to evaluate the quality of a more global region, our multi-view quality metric is able to faithfully reflect the Janus problem within the generated 3D scenes. This mechanism is illustrated in Fig. 4.

We perform two evaluations to investigate the discrepancy in scores for the Janus problem. The first is by randomly selecting 30 pairs of results generated by two different methods using the same prompts and with similar texture quality (that is indistinguishable upon human examination), meanwhile one with and the other without the Janus problem. Secondly, we take 15 generated scenes without the Janus problem and artificially synthesized scenes with the Janus problem by rotating counterparts and fusing them. For both cases, we compare the changes in the quality metric score before and after applying regional convolution. The results in Table 3 show a clear discrepancy in scores, especially after applying regional convolution.

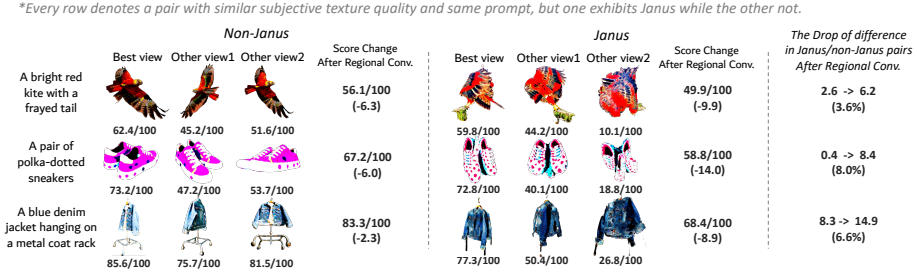
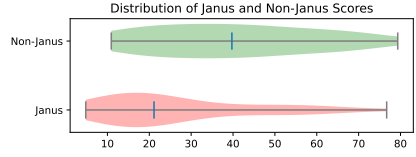


Fig. 4: Underlying mechanism of how our multi-view quality metric reflects the Janus problem: Scores for illed-views are penalized, and regional convolution propagates this drop in local score to the global score.

Table 3: Relative quality score drop from 3D scenes without Janus problem to scenes with Janus problem.

	Relative score drop w/o regional conv	Relative score drop w/ regional conv
Randomly Selected Pairs	2.93%	**4.45%**
Artificially Modified Pairs	4.05%	**7.01%**

Fig. 5: The quality score (normalized to 0-100) distribution of generated 3D scenes with and without Janus problem.



We also present the quality score trend on randomly selected 70 non-collapsed meshes and categorize them based on the presence of the Janus Problem (shown in Fig. 5). We observe that 3D scenes with the Janus Problem are generally scored lower.

4.2 Benchmarking Results

Experimental Setup. Following the prompt generation scheme outlined in Sec. 3.1 and taking into consideration both experimental breadth and test speed, we utilize GPT-4 to generate $N = 100$ prompts for each of the three categories: *single object*, *single object with surroundings*, and *multiple objects*, resulting in a total of 300 prompts. We employ the implementation provided by ThreeStudio [7] (or its extensions) to uniformly evaluate 10 prevalent text-to-3D methods on these prompts, including DreamFusion [27], Magic3D [16], LatentNeRF [22], Fantasia3D [4], SJC [43], ProlificDreamer [44], MVDream [40],

DreamGaussian [41], GeoDream [21], and RichDreamer [29]. We normalize the original scores on quality and alignment assessment from the range $[-2.5, 2.5]$, $[1, 5]$ to $[0, 100]$. We set the five focal lengths used for multi-focal capturing to 3.0, 4.0, 5.0, 6.0, 7.5, and set the resolution of the rendered image to 512×512. When capturing the 3D mesh, we directly use the diffuse color of the texture without additional light source at the corresponding direction as the rendering result. All experiments are conducted on an NVIDIA A100-80GB GPU. We will continuously review the latest methods and update our leaderboard.

To obtain an optimal mesh extraction, Marching Cubes is utilized for DreamFusion, LatentNeRF, ProlificDreamer and MVDream, while other methods employ DMTet; To retain quality without excessive UV unwrapping times, textures are extracted following mesh geometry simplification to a maximum of 40,000 faces. For methods that yield 3D scenes with a diffusion latent radiance field representation rather than RGB, we also convert them into a latent texture map. Subsequently, we transform these into RGB textures using a latent decoder with a sliding window strategy to achieve anti-aliasing conversion.

Results. Tab. 4 reports the quality scores, alignment scores, and the average scores for each text-to-3D method on the three prompt sets in T^{3}Bench. We also showcase some examples in Sec. 11. We summarize our key findings on current methods in the following.

1. A simple combination of SDS and Stable Diffusion can cause density collapse. For **DreamFusion**, the randomness inherent in Stable Diffusion's 2D guidance makes the direct application of Score Distillation Sampling (SDS) to supervise NeRF generation somewhat unstable. This can occasionally result in an inability to form effective density information during the optimization process, leading to failures that lower the overall score. **Magic3D** introduced an additional mesh refinement stage with high-resolution guidance, significantly improving the generated content's quality. However, the first stage still suffers from a high failure rate. **LatentNeRF** reduces this rate and boosts performance by optimizing in the latent domain rather than the RGB domain from the outset. Furthermore, compared to SDS, Score Jacobian Chaining (**SJC**) is less likely to lead to density collapse but tends to produce a large volume of sparse and floating density, making it difficult to extract high-quality meshes and reducing its practicality, as evidenced by our metrics.

2. The Efficiency of Current Text-to-3D Methods Requires Enhancement. Current optimization approaches based on SDS typically necessitate thirty minutes or more, constraining further applications. **DreamGaussian** accelerates content generation by employing Gaussian splatting as scene representation instead of NeRF. However, the meshes extracted from Gaussians often suffer from poor quality, excessive smoothness, and disordered textures, leading to a need for overall performance improvement. Future research into Gaussian splatting techniques and text-to-3D framework with feed-forward design may hold the key to significantly boosting efficiency.

3. VSD achieves rich detail generation at the expense of efficiency and Janus faces. Variational Score Distillation (VSD) proposed by **Prolific-**

Table 4: The average scores of text-to-3D methods on T^{3}Bench.

	Running Time ↓	Quality ↑	Alignment ↑	Average ↑
Single Object				
Dreamfusion [27]	30min	24.9	24.0	24.4
Magic3D [16]	40min	38.7	35.3	37.0
LatentNeRF [22]	65min	34.2	32.0	33.1
Fantasia3D [4]	45min	29.2	23.5	26.4
SJC [43]	25min	26.3	23.0	24.7
ProlificDreamer [44]	240min	51.1	47.8	49.4
MVDream [40]	30min	53.2	42.3	47.8
DreamGaussian [41]	7min	19.9	19.8	19.8
GeoDream [21]	400min	48.4	33.8	41.1
RichDreamer [29]	70min	57.3	40.0	48.6
Single Object with Surroundings				
Dreamfusion [27]	30min	19.3	29.8	24.6
Magic3D [16]	40min	29.8	41.0	35.4
LatentNeRF [22]	65min	23.7	37.5	30.6
Fantasia3D [4]	45min	21.9	32.0	27.0
SJC [43]	25min	17.3	22.3	19.8
ProlificDreamer [44]	240min	42.5	47.0	44.8
MVDream [40]	30min	36.3	48.5	42.4
DreamGaussian [41]	7min	10.4	17.8	14.1
GeoDream [21]	400min	35.2	34.5	34.9
RichDreamer [29]	70min	43.9	42.3	43.1
Multiple Objects				
Dreamfusion [27]	30min	17.3	14.8	16.1
Magic3D [16]	40min	26.6	24.8	25.7
LatentNeRF [22]	65min	21.7	19.5	20.6
Fantasia3D [4]	45min	22.7	14.3	18.5
SJC [43]	25min	17.7	5.8	11.7
ProlificDreamer [44]	240min	45.7	25.8	35.8
MVDream [40]	30min	39.0	28.5	33.8
DreamGaussian [41]	7min	12.3	9.5	10.9
GeoDream [21]	400min	34.3	16.5	25.4
RichDreamer [29]	70min	34.8	22.0	28.4

Dreamer optimizes the distribution of 3D scenes, showing significant benefits in generating detailed information across both single-object settings and complex prompts. Nonetheless, the use of VSD can sometimes lead to the introduction of extraneous details or geometric noise, adversely affecting human perception and BLIP captioning accuracy. This issue becomes more pronounced as the object count increases, leading to a decrease in alignment metrics. Additionally, VSD's modifications do not incorporate 3D or multi-view priors, allowing the persistence of the Janus problem in the generated outcomes. Employing appropriate geometry initialization may help mitigate these issues.

4. Geometry initialization techniques need improvement. Implementing effective geometry initialization before optimization shows the potential to improve 3D content generation. While **Fantasia3D** excels in generating rich textures, its efficacy diminishes in complex scenes due to the less precise geom-

etry it produces, as the supervision of geometry generation only through Stable Diffusion. **GeoDream**, on the other hand, generates a set of pseudo-multi-view images through models like MVDream and Zero123++ [39] to initialize cost volumes, leading to more accurate geometry initialization. However, constrained by the performance of these models, inconsistencies may arise among the multi-view images, resulting in initialization failures (approximately one-fifth based on our benchmarks), highlighting the need for further performance improvement.

5. Leveraging multi-view diffusion models achieving commendable outcomes yet faces challenges with OOD problems. The multi-view diffusion model introduced by **MVDream** demonstrates substantial quality improvements, as reflected in the scores. It also effectively solves the multi-view inconsistency problem that arises with other methods. **RichDreamer** further developed a multi-view diffusion model that incorporates depth, normal, and albedo information. Nevertheless, these methods encounter a limitation in more complex scenarios, where there is a tendency to omit certain elements of the object or environment, or to generate inaccurate colors in the outcomes. This may stem in part from the fact that multi-view diffusion was trained on Objaverse [6], a 3D dataset comprised primarily of centered objects, which explains the struggle with certain out-of-distribution (OOD) cases.

Trends across different prompt sets. As shown in Tab. 4, the overall performance is relatively good for the *Single Object* set, particularly for ProlificDreamer, Magic3D, and MVDream. However, when additional surrounding information is incorporated or when multiple objects are placed, the quality metrics for all methods experience varying degrees of degradation.

In terms of alignment, some methods are able to reflect object information beyond the surroundings. This results in no significant decline in the *Single Object with Surroundings* set compared to the *Single Object* set. However, a noticeable decline is observed when the prompt set changes to *Multiple Objects*. This trend reflects the current issue with most works using Score Distillation Sampling (SDS) as guidance to supervise the generation of 3D scenes. Specifically, SDS is relatively stable for single objects, but when the descriptions of the surroundings are added or when there are multiple objects in the scene, the appearance of the surroundings may have many possibilities after denoising steps. There may be more possibilities for relative positions between multiple objects, leading to increased variability in the results generated by the diffusion model. This in turn reduces the stability when supervising the generation of 3D scenes, resulting in a significant decline in the results.

In contrast, ProlificDreamer uses Variational Score Distillation (VSD) instead of SDS. By optimizing the distribution of the scene rather than directly optimizing the rendering results of the scene for 3D generation, ProlificDreamer demonstrates a clear advantage in complex scenarios. The multi-view diffusion guidance used by MVDream also shows superior performance on multi-object scenes. MVDream tends to generate clearer and more favorable 3D shapes compared to ProlificDreamer, which can sometimes produce redundant densities. However, when dealing with out-of-distribution text prompts outside of the Ob-

javerse dataset, MVDream sometimes struggles to fully capture all of the information from the text (e.g. generate a single object when the text refers to multiple ones).

Parallels and contrasts of the quality and alignment metrics. It is worth noting that quality and alignment are not entirely correlated. Quality is more concerned with the geometry and subjective quality within a certain range, while alignment focuses on accurately restoring the information in the prompt. It is relatively sensitive to additional erroneous information, encouraging the generation of precise and clear 3D scenes. For instance, the overall performance of Fantasia3D decreases markedly when generating multiple objects, as it fails to create precise 3D geometry, resulting in poor alignment compared to LatentNeRF. However, the quality of some generated objects is commendable with the obtained rich texture, making the overall quality higher than LatentNeRF.

ProlificDreamer typically generates more realistic textures, contributing to its superior quality. However, it sometimes generates a large amount of information not mentioned in the prompt, resulting in the possibility that the information described in the prompt only occupies a small part of the 3D generation results. Sometimes it only appears in the form of partial texture without significant geometry, which reduces its alignment index. Moreover, this characteristic is not what subsequent applications of text-to-3D want to see, further highlighting the importance of the alignment metric.

2D Guidance Analysis. In Sec. 10, we investigate the effectiveness of 2D guidance from Stable Diffusion in generating 3D scenes by examining the correlation between the quality of 2D image generation and the quality of resulting 3D scenes. Results show that while Stable Diffusion produces high-quality 2D images, the ability of text-to-3D methods to utilize this guidance for accurate 3D scene generation is limited, reflected in generally low Spearman correlation between 2D image quality and 3D scene quality. The findings highlight that the main challenges in text-to-3D generation are learning 3D structures from 2D guidance and ensuring view consistency.

5 Conclusion

In this work, we present T^3Bench, the first comprehensive benchmark for evaluating text-to-3D generation methods. T^3Bench serves as a rich testbed as it provides diverse prompt suites, and supports fully automatic evaluation by incorporating our proposed multi-view quality and alignment metrics that closely correlate with human judgments. We carefully benchmark 10 prevalent text-to-3D methods on T^3Bench, and highlight a number of common and specific problems with current methods.

6 Discussion

Size of Data. Unlike existing text-to-image methods that enable efficient generation, the current text-to-3D techniques are considerably slower, requiring a

minimum of half an hour and potentially several hours for a single prompt. This makes it hard to test with larger sets of prompts.

Indirect Evaluation. Given the absence of an effective evaluation method that directly aligns the generated 3D scenes with human evaluation, there is an inevitable loss of information during the 3D to 2D rendering process, even with the efficacy of our multi-view capturing and processing scheme in evaluating geometry and other information. Likewise, no 3D captioning framework matches the performance of BLIP in 2D image captioning. While our multi-view captioning and merging strategy typically generates accurate 3D captions, the merging process does not always yield flawless results.

References

1. Bai, Y., Ying, J., Cao, Y., Lv, X., He, Y., Wang, X., Yu, J., Zeng, K., Xiao, Y., Lyu, H., et al.: Benchmarking foundation models with language-model-as-an-examiner. arXiv preprint arXiv:2306.04181 (2023)
2. Bakr, E.M., Sun, P., Shen, X., Khan, F.F., Li, L.E., Elhoseiny, M.: Hrs-bench: Holistic, reliable and scalable benchmark for text-to-image models. arXiv preprint arXiv:2304.05390 (2023)
3. Brooks, T., Holynski, A., Efros, A.A.: Instructpix2pix: Learning to follow image editing instructions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18392–18402 (2023)
4. Chen, R., Chen, Y., Jiao, N., Jia, K.: Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. arXiv preprint arXiv:2303.13873 (2023)
5. Cho, J., Zala, A., Bansal, M.: Dall-eval: Probing the reasoning skills and social biases of text-to-image generation models (2023)
6. Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., Farhadi, A.: Objaverse: A universe of annotated 3d objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13142–13153 (2023)
7. Guo, Y.C., Liu, Y.T., Shao, R., Laforte, C., Voleti, V., Luo, G., Chen, C.H., Zou, Z.X., Wang, C., Cao, Y.P., Zhang, S.H.: threestudio: A unified framework for 3d content generation. `https://github.com/threestudio-project/threestudio` (2023)
8. He, Y., Wang, P., Hu, Y., Zhao, W., Yi, R., Liu, Y.J., Wang, W.: Mmpi: a flexible radiance field representation by multiple multi-plane images blending (2023)
9. Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., Cohen-Or, D.: Prompt-to-prompt image editing with cross attention control. arXiv preprint arXiv:2208.01626 (2022)
10. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
11. Hong, S., Ahn, D., Kim, S.: Debiasing scores and prompts of 2d diffusion for robust text-to-3d generation. arXiv preprint arXiv:2303.15413 (2023)
12. Jain, A., Mildenhall, B., Barron, J.T., Abbeel, P., Poole, B.: Zero-shot text-guided object generation with dream fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 867–876 (2022)

13. Lee, T., Yasunaga, M., Meng, C., Mai, Y., Park, J.S., Gupta, A., Zhang, Y., Narayanan, D., Teufel, H., Bellagente, M., et al.: Holistic evaluation of text-to-image models. Advances in Neural Information Processing Systems **36** (2024)
14. Li, J., Tan, H., Zhang, K., Xu, Z., Luan, F., Xu, Y., Hong, Y., Sunkavalli, K., Shakhnarovich, G., Bi, S.: Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. arXiv preprint arXiv:2311.06214 (2023)
15. Li, J., Li, D., Xiong, C., Hoi, S.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: International Conference on Machine Learning. pp. 12888–12900. PMLR (2022)
16. Lin, C.H., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., Kreis, K., Fidler, S., Liu, M.Y., Lin, T.Y.: Magic3d: High-resolution text-to-3d content creation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 300–309 (2023)
17. Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: Text summarization branches out. pp. 74–81 (2004)
18. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. pp. 740–755. Springer (2014)
19. Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. In: Seminal graphics: pioneering efforts that shaped the field, pp. 347–353 (1998)
20. Luo, T., Rockwell, C., Lee, H., Johnson, J.: Scalable 3d captioning with pretrained models. arXiv preprint arXiv:2306.07279 (2023)
21. Ma, B., Deng, H., Zhou, J., Liu, Y.S., Huang, T., Wang, X.: Geodream: Disentangling 2d and geometric priors for high-fidelity and consistent 3d generation. arXiv preprint arXiv:2311.17971 (2023)
22. Metzer, G., Richardson, E., Patashnik, O., Giryes, R., Cohen-Or, D.: Latent-nerf for shape-guided generation of 3d shapes and textures (2022)
23. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)
24. Mohammad Khalid, N., Xie, T., Belilovsky, E., Popa, T.: Clip-mesh: Generating textured meshes from text using pretrained image-text models. In: SIGGRAPH Asia 2022 conference papers. pp. 1–8 (2022)
25. OpenAI: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
26. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. arXiv preprint arXiv:2209.14988 (2022)
27. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. In: The Eleventh International Conference on Learning Representations (2023)
28. Qian, G., Mai, J., Hamdi, A., Ren, J., Siarohin, A., Li, B., Lee, H.Y., Skorokhodov, I., Wonka, P., Tulyakov, S., et al.: Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. arXiv preprint arXiv:2306.17843 (2023)
29. Qiu, L., Chen, G., Gu, X., Zuo, Q., Xu, M., Wu, Y., Yuan, W., Dong, Z., Bo, L., Han, X.: Richdreamer: A generalizable normal-depth diffusion model for detail richness in text-to-3d. arXiv preprint arXiv:2311.16918 (2023)
30. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from

natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
31. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
32. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
33. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E., Ghasemipour, S.K.S., Ayan, B.K., Mahdavi, S.S., Lopes, R.G., Salimans, T., Ho, J., Fleet, D.J., Norouzi, M.: Photorealistic text-to-image diffusion models with deep language understanding (2022)
34. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. Advances in Neural Information Processing Systems **35**, 36479–36494 (2022)
35. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al.: Laion-5b: An open large-scale dataset for training next generation image-text models. Advances in Neural Information Processing Systems **35**, 25278–25294 (2022)
36. Seo, H., Kim, H., Kim, G., Chun, S.Y.: Ditto-nerf: Diffusion-based iterative text to omni-directional 3d model. arXiv preprint arXiv:2304.02827 (2023)
37. Seo, J., Jang, W., Kwak, M.S., Ko, J., Kim, H., Kim, J., Kim, J.H., Lee, J., Kim, S.: Let 2d diffusion model know 3d-consistency for robust text-to-3d generation. arXiv preprint arXiv:2303.07937 (2023)
38. Shen, T., Gao, J., Yin, K., Liu, M.Y., Fidler, S.: Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. Advances in Neural Information Processing Systems **34**, 6087–6101 (2021)
39. Shi, R., Chen, H., Zhang, Z., Liu, M., Xu, C., Wei, X., Chen, L., Zeng, C., Su, H.: Zero123++: a single image to consistent multi-view diffusion base model. arXiv preprint arXiv:2310.15110 (2023)
40. Shi, Y., Wang, P., Ye, J., Long, M., Li, K., Yang, X.: Mvdream: Multi-view diffusion for 3d generation. arXiv preprint arXiv:2308.16512 (2023)
41. Tang, J., Ren, J., Zhou, H., Liu, Z., Zeng, G.: Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. arXiv preprint arXiv:2309.16653 (2023)
42. Tsalicoglou, C., Manhardt, F., Tonioni, A., Niemeyer, M., Tombari, F.: Textmesh: Generation of realistic 3d meshes from text prompts. arXiv preprint arXiv:2304.12439 (2023)
43. Wang, H., Du, X., Li, J., Yeh, R.A., Shakhnarovich, G.: Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12619–12629 (2023)
44. Wang, Z., Lu, C., Wang, Y., Bao, F., Li, C., Su, H., Zhu, J.: Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. arXiv preprint arXiv:2305.16213 (2023)
45. Xu, J., Wang, X., Cheng, W., Cao, Y.P., Shan, Y., Qie, X., Gao, S.: Dream3d: Zero-shot text-to-3d synthesis using 3d shape prior and text-to-image diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20908–20918 (2023)

46. Xu, J., Liu, X., Wu, Y., Tong, Y., Li, Q., Ding, M., Tang, J., Dong, Y.: Imagereward: Learning and evaluating human preferences for text-to-image generation. arXiv preprint arXiv:2304.05977 (2023)
47. Yi, T., Fang, J., Wu, G., Xie, L., Zhang, X., Liu, W., Tian, Q., Wang, X.: Gaussiandreamer: Fast generation from text to 3d gaussian splatting with point cloud priors. arXiv preprint arXiv:2310.08529 (2023)
48. Zhang, K., Riegler, G., Snavely, N., Koltun, V.: Nerf++: Analyzing and improving neural radiance fields. arXiv preprint arXiv:2010.07492 (2020)
49. Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q., Artzi, Y.: Bertscore: Evaluating text generation with bert. arXiv preprint arXiv:1904.09675 (2019)
50. Zhao, M., Zhao, C., Liang, X., Li, L., Zhao, Z., Hu, Z., Fan, C., Yu, X.: Efficientdreamer: High-fidelity and robust 3d creation via orthogonal-view diffusion prior. arXiv preprint arXiv:2308.13223 (2023)

7 Example Prompts

7.1 Question Generation

Single Object.

Please describe 20 objects' appearance for me in brief words, without background. Please make sure that the object you provided has enough diversity, and that the format is similar to my example. Here is an example: "A pig wearing a backpack".

Single Object with Surroundings.

Please describe 20 objects for me in brief words. Please make sure that the object you provided has enough diversity, and that the format is similar to my example. Here is an example: "A black metal bicycle leaning against a brick wall".

Multiple Objects.

Please describe 20 different scenes for me in brief words, each scene contains multiple objects. Do not describe the environment. Please make sure that the scenes you provided have enough diversity, and the format similar to my example. Here are examples: "A child with a red shirt is playing with a dog", or "Two coffee cups stand on the table".

7.2 Multiple Caption Merging

Given a set of descriptions about the same 3D object, distill these descriptions into one concise caption. The descriptions are as follows:
view1: ...
view2: ...
...
view{N}: ...
Avoid describing background, surface, and posture. The caption should be:

7.3 LLM Likert Scale Scoring

You are an assessment expert responsible for prompt-prediction pairs. Your task is to score the prediction according to the following requirements:
1. Evaluate the recall, or how well the prediction covers the information in the prompt. if the prediction contains information that does not appear in the prompt, it should not be considered as bad.
2. If the prediction contains correct information about color or features in the prompt, you should also consider raising your score.
3. Assign a score between 1 and 5, with 5 being the highest. Do not provide a complete answer; give the score in the format: 3
Prompt: ...
Prediction: ...

Fig. 6: Schema of icosahedrons with different levels.

8 Experimental Details

8.1 Metric Evaluation

For the evaluation of metrics, we randomly select 30% of the prompts from each prompt set, along with their corresponding 3D mesh generated by the text-to-3D method. This results in a total of 630 samples. We request human annotators to carefully check the mesh in an interactive 3D viewer and score the responses on a scale of 1-5, based on their 3D quality and alignment. Below, we provide the annotation instructions:

> 1. Scoring is based on two dimensions: quality (which assesses the subjective quality of the 3D generation) and alignment (which evaluates how well the generated content covers the original prompt). These two dimensions are scored on a scale of 1 to 5, with 1 being the lowest and 5 the highest.
> 2. Please drag each generated mesh to our specified 3D viewer. After carefully examining the mesh from various angles, assign your score based on the above two dimensions.

8.2 Capture Viewpoint Selection

In order to uniformly select the capturing location of the 3D mesh, we construct the icosahedron and use its vertices as the location for the captures. The vertex coordinates of a level-0 unit icosahedron are computed as follows:

$$V^{(0)} = \sqrt{1+\phi^2} \cdot \begin{vmatrix} \phi & 1 & 0 \\ -\phi & 1 & 0 \\ \phi & -1 & 0 \\ -\phi & -1 & 0 \\ 1 & 0 & \phi \\ 1 & 0 & -\phi \\ -1 & 0 & \phi \\ -1 & 0 & -\phi \\ 0 & \phi & 1 \\ 0 & -\phi & 1 \\ 0 & \phi & -1 \\ 0 & -\phi & -1 \end{vmatrix}, \tag{2}$$

where

$$\phi = \frac{1+\sqrt{5}}{2}, \tag{3}$$

and there is an edge between every two points with a distance of $2/\sqrt{1+\phi^2}$, resulting in 12 vertices, 30 edges, and 20 triangle faces.

A level-K unit icosahedron can be obtained recursively by adding an extra vertice on every edge of a level-(K-1) unit icosahedron and adding an edge between every two new vertexes with a triangle face of the level-(K-1) unit icosahedron, then scaling every new vertex's coordinate to a length of 1. A demonstration of different level icosahedrons is shown in Fig. 6.

8.3 Capturing Poses Derivation

Since many text image scoring models are sensitive to rotation, we need to make sure that the angle of the shot is as free as possible from 2D rotation around the look-at vector. We ensure this constraint with the following procedure:

Given the location v of the shot, we can get the look-at vector as follows:

$$\mathbf{l} = -\frac{\mathbf{v}}{||\mathbf{v}||}. \tag{4}$$

Then, we acquire the horizontal vector $\mathbf{r}$ of the camera plane by

$$\mathbf{r} = \frac{\mathbf{u} \times \mathbf{l}}{||\mathbf{u} \times \mathbf{l}||}, \tag{5}$$

where $\mathbf{u}$ is the unit vector parallel with the positive direction of the vertical axis. The up vector of the camera plane can be calculated by

$$\mathbf{u}' = \mathbf{l} \times \mathbf{r}. \tag{6}$$

Finally, the camera matrix $\mathbf{P}$ is formed with

$$\mathbf{P} = [-\mathbf{r} \quad \mathbf{u}' \quad \mathbf{l} \quad \mathbf{v}]. \tag{7}$$

9 Design Choices for Regional Convolution

The general form of regional convolution can be formed as:

$$s_i^{(t+1)} = \frac{1}{w|N(i)|+1}\left(s_i^{(t)} + w\sum_{j\in N(i)} s_j^{(t)}\right), \tag{8}$$

To further explore the use of convolution kernels and the impact of the receptive field on the quality evaluation metric, we conduct experiments on correlations with human annotations, where the convolution weights w are varied from 0 to 2, and different times of convolutions are applied.

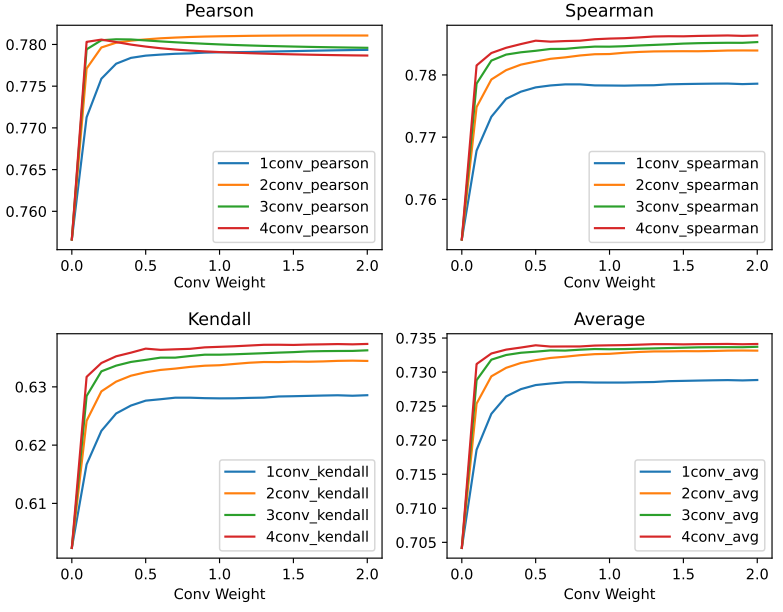


Fig. 7: Correlation variations on different weights and times of convolutions.

We observe that as the number of convolutions and the weights of neighbors in convolutions increases, both the Spearman and Kendall correlation coefficients consistently rise. However, the Pearson correlation coefficient exhibits a more complex trend. Specifically, it decreases when the number of convolutions and the weights of neighboring convolutions are excessively high. This phenomenon could be attributed to a smoother convolution operation with a larger reception field, which takes a more overall sense, is more beneficial for preserving the order of evaluation metrics. However, excessive smoothing can overlook finer details such as texture quality, leading to a non-uniform compression of scores and a reduction in the linearity of the quality evaluation metric. Considering these factors, we ultimately selected a weight $w = 1$ and applied three times of convolutions.

10 2D Guidance Analysis

The majority of current text-to-3D methods utilize 2D priors associated with Stable Diffusion [31] for the generation of 3D scenes. To delve deeper into the effectiveness of 2D guidance and the capabilities of current text-to-3D methods in utilizing this guidance for 3D generation, we explore the correlation between the quality of 2D image generation produced by the diffusion model and the resulting quality of the 3D generation. For each prompt in T^{3}Bench, we apply the Stable Diffusion backbone of each method for text-to-image generation. Notably, the text-to-3D methods utilize view-dependent prompting in conjunction with 2D guidance from the diffusion model during the generation process. Descriptions of viewing angles (e.g. front view, side view) are added at the end of the prompt. Given that the range and granularity of viewing descriptions in view-dependent prompting vary across different text-to-3D methods, we directly use the original prompt without view-dependent prompting in the text-to-image generation. We then compute the single-view quality metric on the generated 2D image. Finally, we compute the correlation between the single-view quality metric of the generated 2D image and the quality metric (Multi-view capturing with ImageReward) result of the generated 3D scene.

Tab. 5 displays the Spearman correlation between the text-to-image scores for the 2D guidance and the final text-to-3D scores. It can be observed that that all correlations are relatively low, and there are two overall trends: 1) methods demonstrating better performance in text-to-3D also have higher correlation coefficients; and 2) when using different prompt sets, the correlation coefficient also follows the trend of *Single Object* greater than *Single Object with Surroundings*, and the latter greater than *Multiple Objects*. We attribute these outcomes to the fact that Stable Diffusion can generate satisfactory 2D images most of the time, even for complex prompts. However, 2D guidance may not be effectively used by text-to-3D methods — they may fail to generate accurate 3D scenes even though the 2D images are acceptable, leading to a low text-to-3D score while high text-to-image score. In addition, the 2D guidance may not be view-consistent, which does not significantly affect the text-to-image scores but can

Table 5: The Spearman's ρ correlation between the text-to-3D methods' generation qualities and the diffusion models' 2D image generation qualities, averaged over all prompts.

	Single Obj.	Single Obj. with Surr.	Multi Obj.
Dreamfusion	0.211	0.184	0.045
Magic3D	0.229	0.158	0.059
LatentNeRF	0.290	0.191	0.050
Fantasia3D	0.159	0.153	0.006
SJC	0.228	0.159	0.040
ProlificDreamer	0.357	0.272	0.147
MVDream	0.421	0.340	0.474
DreamGaussian	0.206	0.132	0.156
GeoDream	0.330	0.228	0.086
Richdreamer	0.407	0.347	0.252

indeed lead to poorer quality in the final 3D generation. Superior methods like ProlificDreamer can better utilize 2D images to form a 3D scene, as suggested by its higher correlation, and as a result, can generate higher quality 3D scenes.

The retrained multi-view diffusion model by MVDream (also leveraged by RichDreamer) provides effective guidance for 3D generation, as evidenced by the highest correlation results. This highlights the capabilities of the retrained diffusion model. However, the retrained diffusion itself exhibits a degree of degradation in its generation capabilities, especially in scenarios involving surrounding information and multiple objects. This is reflected in the lower average scores of 2D image generation with MVDream's diffusion model compared to Stable Diffusion (e.g. 32.9 vs 44.0 on the multi-object set). While the retrained diffusion model is useful for guiding 3D generation, there is still room for improvement in diffusion modeling for 3D tasks.

These observations suggest that the current bottleneck of text-to-3D lies in the process of learning 3D from 2D guidance and the view consistency of 2D guidance, rather than the generative capability of Stable Diffusion itself.

11 More Case Studies

11.1 Single-view vs. Multi-view Capturing

We further illustrate through a case study that adhering to the previous method and only capturing single-view images does not yield satisfactory evaluations. As depicted in Fig. 8, the first two examples demonstrate good subjective quality in the front view. However, their geometries are relatively poor, and there are noticeable residuals or artifacts when they are converted to other viewpoints. These can be identified with our multi-view capturing mechanism, which subsequently adjusts the scores accordingly. In the next two examples, the front view is partially obscured, which fails to fully represent the subjective quality of

Fig. 8: Comparisons of the scoring between single-view capturing and our multi-view capturing scheme. The first image column denotes the single front view, and the other two image columns are captured from other viewpoints.

the generated objects. Our multi-view capturing mechanism can detect this and improve their scores accordingly.

11.2 More results

We provide case studies of test prompts with generations and evaluations of different text-to-3D methods in Figs. 9, 10, 11, 12.

Fig. 9: Visualizations of text-to-3D generation results. The two scores denote quality and alignment, respectively.

Fig. 10: More results of our test prompts, including generations and evaluations of different text-to-3D methods (#1).

Fig. 11: More results of our test prompts, including generations and evaluations of different text-to-3D methods (#2).

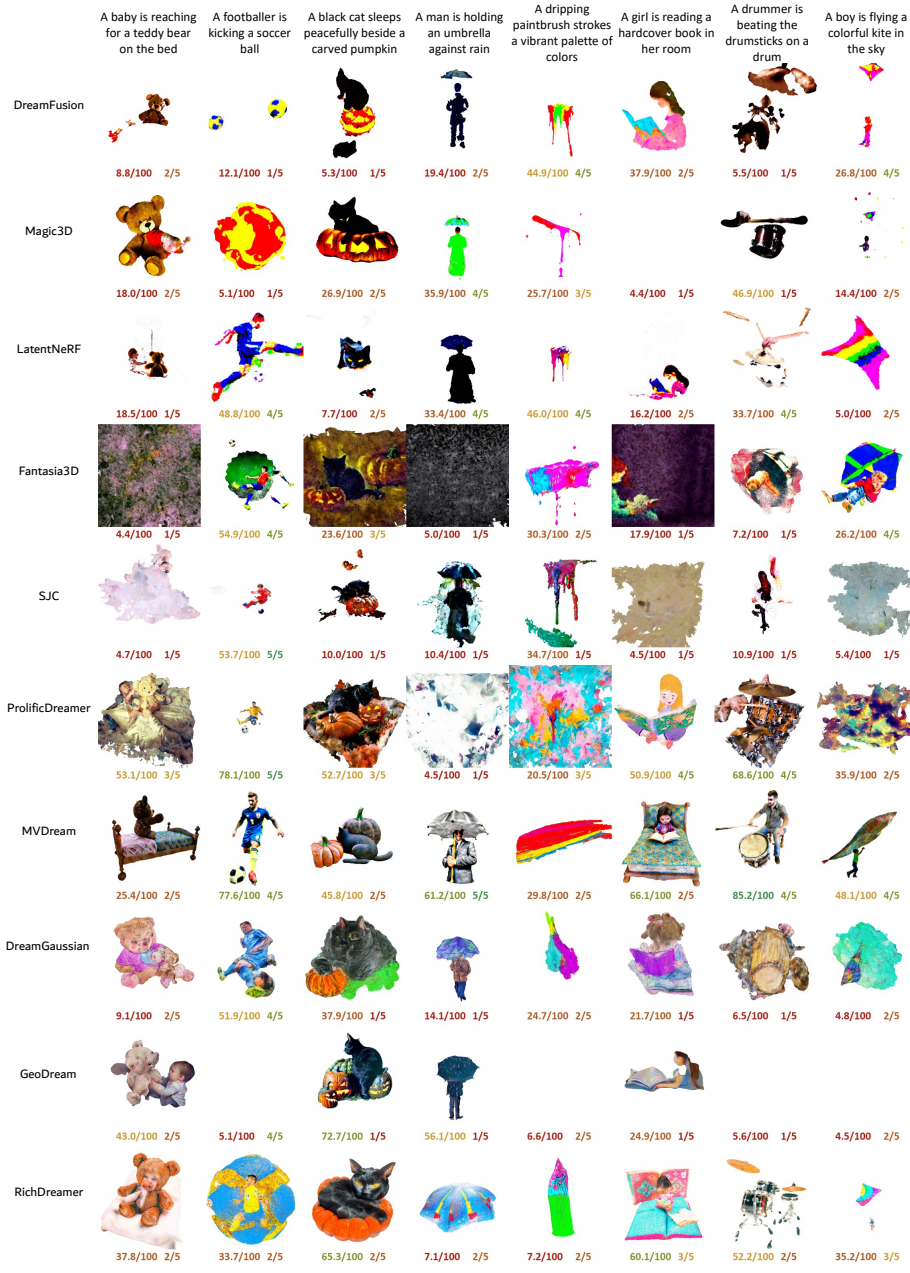


Fig. 12: More results of our test prompts, including generations and evaluations of different text-to-3D methods (#3).