GPT-4V(ision) is a Human-Aligned Evaluator for Text-to-3D Generation

Tong Wu[1,5*] Guandao Yang[2*] Zhibing Li[1,5*] Kai Zhang[3] Ziwei Liu[4]
Leonidas Guibas[2] Dahua Lin[1,5] Gordon Wetzstein[2]
[1] The Chinese University of Hong Kong [2] Stanford University [3] Adobe Research
[4] S-Lab, Nanyang Technological University [5] Shanghai Artificial Intelligence Laboratory

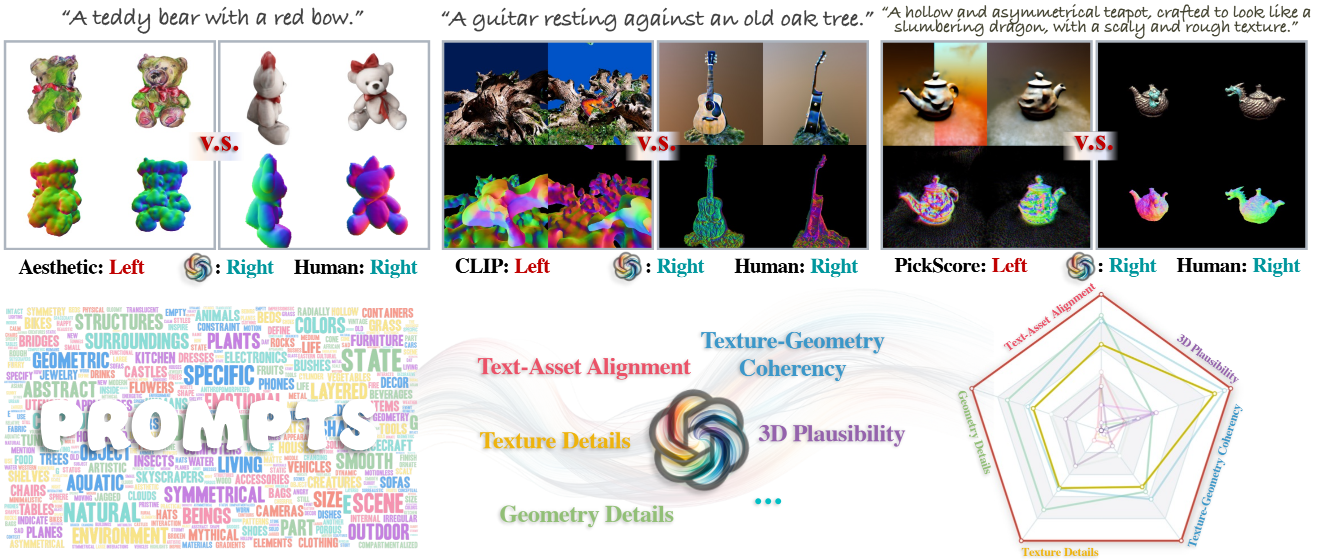


Figure 1. We present a versatile and human-aligned evaluation metric for text-to-3D generative methods. To this end, we design a prompt generator that can produce a set of input prompts targeting an evaluator's demands. Moreover, we leverage GPT-4V to compare two 3D shapes according to different evaluation criteria. Our method provides a scalable and holistic way to evaluate text-to-3D models.

Abstract

Despite recent advances in text-to-3D generative methods, there is a notable absence of reliable evaluation metrics. Existing metrics usually focus on a single criterion each, such as how well the asset aligned with the input text. These metrics lack the flexibility to generalize to different evaluation criteria and might not align well with human preferences. Conducting user preference studies is an alternative that offers both adaptability and human-aligned results. User studies, however, can be very expensive to scale. This paper presents an automatic, versatile, and human-aligned evaluation metric for text-to-3D generative models. To this end, we first develop a prompt generator using GPT-4V to generate evaluating prompts, which serve as input to compare text-to-3D models. We further design a method instructing GPT-4V to compare two 3D assets according to user-defined criteria. Finally, we use these pairwise comparison results to assign these models Elo ratings. Experimental results suggest our metric strongly aligns with human preference across different evaluation criteria. Our code is available at https://github.com/3DTopia/GPTEval3D.

1. Introduction

The field of text-to-3D generative methods has seen remarkable progress over the past year, driven by a series of breakthroughs. These include advancements in neural 3D representations [41, 46], the development of extensive datasets [10, 13, 14], the emergence of scalable generative models [23, 55, 61], and the innovative application of text–image foundational models for 3D generation [47, 50]. Given this momentum, it's reasonable to anticipate rapidly increasing research efforts and advancements within the realm of text-to-3D generative models.

Despite recent advances, the development of adequate evaluation metrics for text-to-3D generative models has not

[*] Equal contribution.

kept pace. This deficiency can hinder progress in further improving these generative models. Existing metrics often focus on a single criterion, lacking the versatility for diverse 3D evaluation requirements. For instance, CLIP-based metrics [28, 50] are designed to measure how well a 3D asset aligns with its input text, but they may not be able to adequately assess geometric and texture detail. This lack of flexibility leads to misalignment with human judgment in evaluation criteria the metric is not designed for. Consequently, many researchers rely on user studies for accurate and comprehensive assessment. Although user studies are adaptable and can accurately mirror human judgment, they can be costly, difficult to scale, and time-consuming. As a result, most user studies have been conducted on a very limited set of text-prompt inputs. This leads to a question: *Can we create automatic metrics that are versatile for various evaluation criteria and align closely with human judgment?*

Designing metrics that meet these criteria involves three essential capabilities: generating input text prompts, understanding human intention, and reasoning about the three-dimensional physical world. Fortunately, Large Multimodal Models (LMMs), particularly GPT-4Vision (GPT-4V) [45], have demonstrated considerable promise in fulfilling these requirements [70]. Drawing inspiration from humans' ability to perform 3D reasoning tasks using 2D visual information under language guidance, we posit that GPT-4V is capable of conducting similar 3D model evaluation tasks.

In this paper, we present a proof-of-concept demonstrating the use of GPT-4V to develop a customizable, scalable, and human-aligned evaluation metric for text-to-3D generative tasks. Building such an evaluation metric is similar to creating an examination, which requires two steps: formulating the questions and evaluating the answers. To effectively evaluate text-to-3D models, it is crucial to obtain a set of input prompts that accurately reflect the evaluators' needs. Relying on a static, heuristically generated set of prompts is insufficient due to the diverse and evolving nature of evaluator demands. Instead, we developed a "meta-prompt" system, where GPT-4V generates a tailored set of input prompts based on evaluation focus. Following the generation of these input text prompts, our approach involves comparing 3D shapes against user-defined criteria, akin to grading in an exam. We accomplish this through designing an instruction template, which can guide GPT-4V to compare two 3D shapes per user-defined criterion. With these components, our system can automatically rank a set of text-to-3D models by assigning each of these models an Elo rating. Finally, we provide preliminary empirical evidence showing that our proposed framework can surpass existing metrics in achieving better alignment with human judgment in a diverse set of evaluation criteria. Results suggest that our metric can efficiently provide an efficient and holistic evaluation of text-to-3D generative models.

2. Related Work

Text-to-3D generation. Text-to-image generation models have become increasingly powerful with text-to-3D extensions being the next frontier (see [47] for a recent survey). However, due to limited amounts of 3D data, text-to-3D has mainly been driven by methods based on optimizing a NeRF representation [41]. For example, Dreamfusion [50] optimizes a NeRF using score-distillation-sampling-based (SDS) loss. The quality of such optimization-based methods [11, 36, 40, 50, 59, 62, 65, 67], however, is far behind that of text-to-image models [49, 53–55]. Compared with their 2D counterparts, they are generally lacking diversity, texture fidelity, shape plausibility, robustness, speed, and comprehension of complex prompts. On the other hand, Point-E [43] and Shap-E [29] train feed-forward 3D generative models on massive undisclosed 3D data. Though they show promising results with fast text-to-3D inference, their generated 3D assets look cartoonish without geometric and texture details. Recently, we notice a rapid change in the landscape of text-to-3D methods [37, 38] mainly due to the public release of the large-scale Objaverse datasets [15, 16]. Feed-forward methods trained on these datasets, e.g., Instant3D [35], have managed to make a big jump in text-to-3D quality, narrowing the performance gap between 3D and 2D generation. As we expect to see continuing progress in this area, it is critical to have robust evaluation metrics closely aligning with human judgment to measure different aspects of 3D generative models, including shape plausibility and texture sharpness. Such an evaluation metric can provide meaningful guidance for model design choices and support fair comparisons among the research community.

3D Evaluation Metrics. Evaluating 3D generative models is inherently challenging, requiring an understanding of both physical 3D worlds and user intentions. Traditional methods for evaluating unconditional or class-conditioned 3D models typically measure the distance between distributions of generated and reference shapes [1, 5, 9, 20, 39, 69]. However, these metrics are not readily applicable to text-conditioned generative tasks due to the difficulty in obtaining a comprehensive reference set, given the vastness of natural language inputs [6]. To alleviate this issue, prior work tried to curate a set of text prompts to evaluate key aspects of text-conditioned generative tasks [21, 50]. Our work complements this effort by creating a text-prompt generator using language instruction. Additionally, prior studies utilized multimodal embeddings, such as CLIP [51] and BLIP [33, 34], to aid the evaluation. For instance, the CLIP Similarity metric [28, 50] employs CLIP embeddings to assess text-to-3D alignment. However, these metrics are often tailored to measure specific criteria, lacking the flexibility to adapt to different requirements of text-to-3D evaluation. User preference studies are considered the gold standard for evaluating text-to-3D models, as adopted by many papers [5, 25, 36, 52, 57, 62]. While user studies offer versatility and accuracy, they are costly, time-consuming, and difficult to scale. Our automatic metrics can serve as an alternative to user preference studies, aligning well with human preferences while offering high customizability.

Large multimodality models. Following the success of large language models (LLMs) [3, 8, 12, 24, 45, 63], the focus has shifted to large multimodal models (LMMs) as the next frontier in artificial intelligence. Initial efforts of LMM involve combining computer vision with LLMs by fine-tuning visual encoders to align with language embeddings [2, 4, 17, 27, 33, 34, 64] or converting visual information to text [26, 58, 66, 71]. Most of these models are usually limited in scale. Recently, GPT-4V [44] has risen as the leading LMMs, benefiting from training on an unprecedented scale of data and computational resources. These LMMs have demonstrated a range of emerging properties [70], including their capability as evaluators for language and/or vision tasks [22, 73, 74]. In our work, we explore the use of GPT-4V in evaluating 3D generative models, a relatively under-explored application because GPT-4V cannot directly consume 3D information.

3. Method Overview

The goal of our evaluation metric is to rank a set of text-to-3D models based on user-defined criteria. Our method consists of two primary components. First, we need to decide which text prompt to use as input for the evaluation task. Toward this goal, we develop an automatic prompt generator capable of producing text prompts with customizable levels of complexity and creativity (Sec 4). The second component is a versatile 3D assets comparator (Sec 5). It compares a pair of 3D shapes generated from a given text prompt according to the input evaluation criteria. Together, these components allow us to use the Elo rating system to assign each of the models a score for ranking (Sec 5.3).

4. Prompt Generation

Creating evaluation metrics for text-to-3D generative models requires deciding which set of input text prompts we should use as input to these models. Ideally, we would like to use all possible user input prompts, but this is computationally infeasible. Alternatively, we would like to build a generator capable of outputting prompts that can mimic the actual distribution of user inputs. To achieve this, we first outline the important components of an input prompt for text-to-3D models (Sec 4.1). Building on these components, we design a "meta-prompt" to instruct GPT-4V how to leverage these components to generate an input text prompt for text-to-3D models (Sec 4.2).

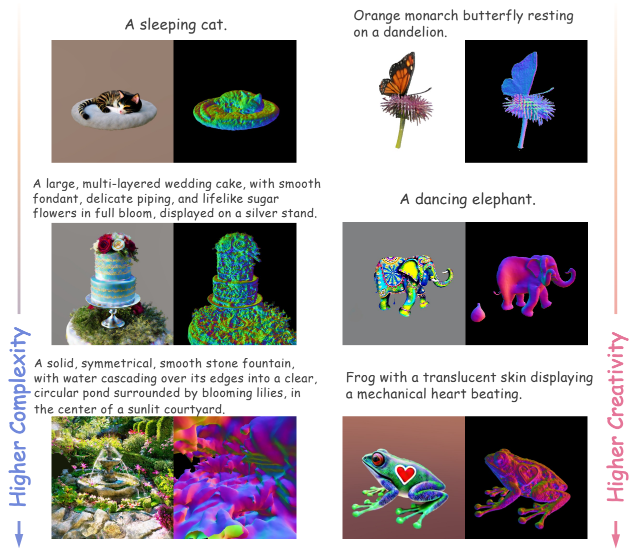

A sleeping cat.
Orange monarch butterfly resting on a dandelion.
A large, multi-layered wedding cake, with smooth fondant, delicate piping, and lifelike sugar flowers in full bloom, displayed on a silver stand.
A dancing elephant.
A solid, symmetrical, smooth stone fountain, with water cascading over its edges into a clear, circular pond surrounded by blooming lilies, in the center of a sunlit courtyard.
Frog with a translucent skin displaying a mechanical heart beating.
Higher Complexity
Higher Creativity

Figure 2. **Controllable prompt generator.** More complexity or more creative prompts often lead to a more challenging evaluation setting. Our prompt generator can produce prompts with various levels of creativity and complexity. This allows us to examine text-to-3D models' performance in different cases more efficiently.

4.1. Prompt components

A typical input text prompt for text-to-3D models contains three components: subjects, properties, and compositions. Subjects usually involve nouns referring to objects or concepts the user would like to instantiate in 3D. "Cats", "fire", and "universe" are all examples of subjects. Properties include adjectives a user can use to describe the subjects or their interactions, such as "mysterious" and "weathered". Finally, users will compose these concepts and properties together into a sentence or clause. The composition varies from as simple as joining different subjects and/or properties together with commas or as thoughtful as writing it as a grammatically correct and fluent sentence. In this work, we prompt GPT-4V to create a comprehensive list of words for subjects and properties. This list of subjects and properties will be used as building blocks to construct the "meta-prompt", which is an instruction for GPT-4V to generate input text-prompts by composing these building blocks. Appendix B.1 contains more implementation details.

4.2. Meta-prompt

Provided with ingredients to create input prompts, we now need to automatically compose these ingredients together according to the evaluator-specified requirements. This requires the prompt generator to understand and follow the evaluator's instruction. In this paper, we use GPT-4V's ability to generate prompts following instructions. Specifically, we would like to build a text instruction asking GPT-4V to create a list of prompts that can be used as input for

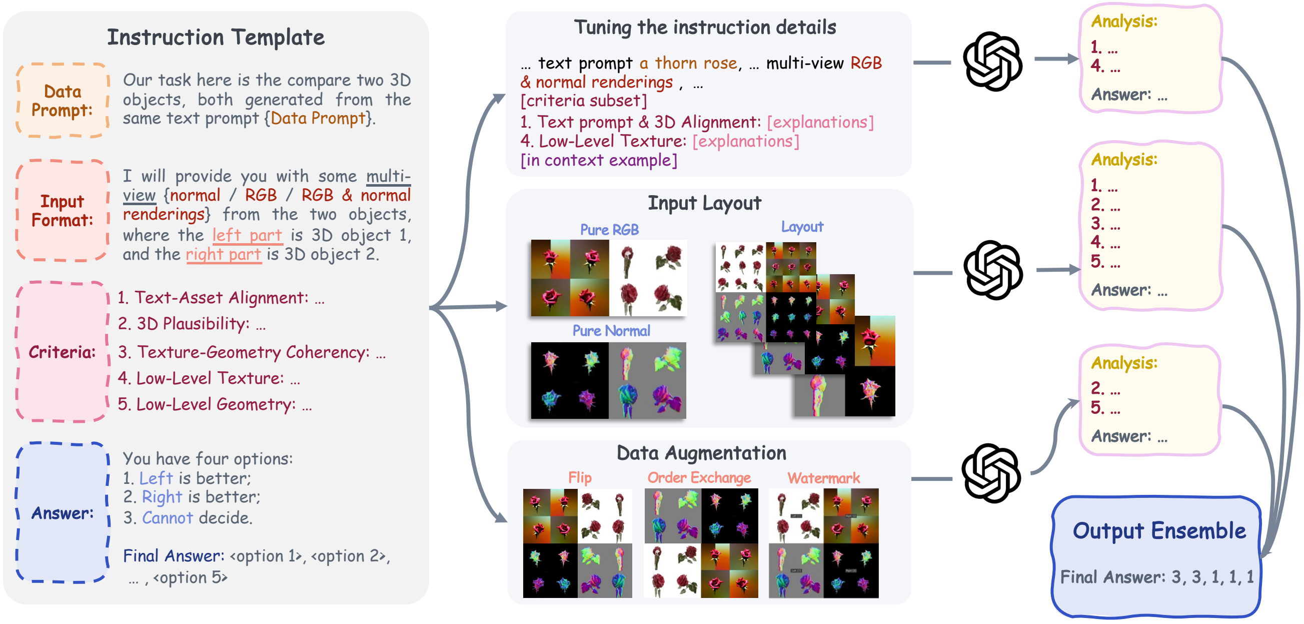

Instruction Template
Data Prompt:
Our task here is the compare two 3D objects, both generated from the same text prompt {Data Prompt}.
Input Format:
I will provide you with some multi-view {normal / RGB / RGB & normal renderings} from the two objects, where the left part is 3D object 1, and the right part is 3D object 2.
Criteria:
1. Text-Asset Alignment: ...
2. 3D Plausibility: ...
3. Texture-Geometry Coherency: ...
4. Low-Level Texture: ...
5. Low-Level Geometry: ...
Answer:
You have four options:
1. Left is better;
2. Right is better;
3. Cannot decide.
Final Answer: <option 1>, <option 2>, ... , <option 5>
Tuning the instruction details
... text prompt a thorn rose, ... multi-view RGB & normal renderings , ...
[criteria subset]
1. Text prompt & 3D Alignment: [explanations]
4. Low-Level Texture: [explanations]
[in context example]
Input Layout
Pure RGB
Layout
Pure Normal
Data Augmentation
Flip
Order Exchange
Watermark
Analysis:
1. ...
4. ...
Answer: ...
Analysis:
1. ...
2. ...
3. ...
4. ...
5. ...
Answer: ...
Analysis:
2. ...
5. ...
Answer: ...
Output Ensemble
Final Answer: 3, 3, 1, 1, 1

Figure 3. **Illustration of how our method compares two 3D assets.** We create a customizable instruction template that contains necessary information for GPT-4V to conduct comparison tasks for two 3D assets (Sec 5.1). We complete this template with different evaluation criteria, input 3D images, and random seeds to create the final 3D-aware prompts for GPT-4V. GPT-4V will then consume these inputs to output its assessments. Finally, we assemble GPT-4V's answers to create a robust final estimate of the task (Sec 5.2)

text-to-3D models. We coin this instruction "meta-prompt".

In order for GPT-4V to output prompts for text-to-3D models, we first provide GPT-4V with the necessary ingredients, *i.e.* a list of subjects and properties from the previous section. In addition to these, the meta-prompt needs to include a description of how the evaluator wants the output prompt set to be. For example, the evaluator might want to focus on complex prompts containing multiple subject interactions and properties, testing a text-to-3D models' ability to generate complex objects. One might also be curious about these models' performance in creative prompts involving subjects and descriptions that are not commonly seen in the real world. How complex and creative the input prompt can influence how challenging the evaluation task is. These two axes, complexity and creativity, are examples of evaluator's criteria. Such criteria should be specified as language instructions attached to the "meta-prompt" along with all the ingredients. With both the prompt ingredient and the evaluator's criteria properly included in the meta-prompt, our GPT-4V-based prompt generator can now compose sentences that adhere to the evaluator's requirement. Appendix B.1 contains more details about our meta-prompt and prompt generation pipeline.

Figure 2 shows prompts outputted from our generator with instruction asking for different complexity and creativity. We can see that high complexity introduces a larger number of objects, multifaceted descriptions, and occasionally, a completely broken scene. Similarly, more creative prompts combine subjects, verbs, or adjectives in unconventional ways. Text-to-3D models also tend to struggle with these more creative prompts, failing to follow the description of these input prompts exactly. This suggests that input prompts distribution can greatly affect how challenging the evaluation task is. Being able to control the distributions of the input prompt allows us to examine the performance of these text-to-3D models through a more focused lens.

5. 3D Assets Evaluator

Now we can sample a set of text prompts, $\mathcal{T} = \{t_i\}_i$, using our generator. In this section, we will evaluate the performance of a set of text-to-3D generative models using $\mathcal{T}$ as input prompts. Given a set of these models, $\mathcal{M} = \{M_j\}_j$, we use each model to generate one or more 3D shapes for each prompt. This results in a set of tuples: $\{(M_k, t_k, M_j(t_k, \mathbf{z}_k)) | M_k \in \mathcal{M}, t_k \in \mathcal{T}\}_k$, where $\mathbf{z}_k$ represents the random noise influencing the shape generation. Our objective is to rank the text-to-3D models in $\mathcal{M}$ based on a user-defined criterion. To accomplish this, we first prompt GPT-4V to compare two 3D assets generated from the same input text prompt (Sec 5.1 and Sec 5.2). We then use these pairwise comparison results to assign each of the models an Elo rating reflecting its performance (Sec 5.3).

5.1. Pairwise Comparison

At the core of our evaluation metric is the ability to answer the following question: *given a text prompt t, and two 3D shapes generated from two different models, say M_i and M_j, which 3D shape is better according to the evaluation criteria?* As discussed in previous sections, we hypothesize that one can leverage GPT-4V to achieve this task. However, since GPT-4V is trained on language and visual data, it lacks the ability to analyze 3D shapes directly. Therefore, our input to GPT-4V should include both text instructions and 2D visual renderings that can capture 3D information.

Specifically, for each of the two 3D assets, we will create a large image containing renderings of the 3D asset from four or nine viewpoints. These two images will be concatenated together before passing into GPT-4V along with the text instructions. GPT-4V will return a decision of which of the two 3D assets is better according to the instruction.

Text instruction. We need to communicate three pieces of information for GPT-4V to compare two 3D assets: instructions to complete a 3D comparison task, the evaluation criteria, and descriptions of the output format. We found it important to emphasize that the provided images are renders from different viewpoints of a 3D object. In addition to a plain description of the user-defined evaluation criteria, providing instruction about what kind of image features one should use when analyzing for a particular criteria is also useful. Finally, instead of requesting only the answer of which shape is better directly, we also prompt GPT-4V to explain how it arrives at its conclusion [7, 68].

Visual features of 3D shapes. Once GPT-4V has been prompted to understand the evaluation criteria and task of interest, we now need to feed the 3D shape into the GPT-4V model. Specifically, we need to create images that can convey the appearance and the geometry features of the 3D shapes. To achieve that, for each 3D object, we create image renders of the object from various viewpoints. For each of these viewpoints, we also render a surface normal image. These normal surface renders will be arranged in the same layout as the RGB render before being fed into GPT-4V. Using world-space surface normal renders leads to better results because they provide geometric information about the surface and allow reasoning for correspondence between views. Appendix B.2 has more implementation details.

5.2. Robust Ensemble

Even though GPT-4V is able to provide an answer to the pairwise shape comparison problem, its response to the same input can vary from time to time due to the probabilistic nature of its inference algorithm. In other words, we can view our GPT-4V 3D shape comparator's outputs as a categorical distribution, and each response is a sample from the distribution. As a result, a single response from GPT-4V might not capture its true prior knowledge since it can be affected by the variance during sampling. This is particularly the case when the variance of the output distribution is high (*e.g.*, when both choices are equally likely). Note that this is not a weakness of GPT-4V as similar situations can happen to human annotators when two objects are equally good according to a criterion. In other words, we are not interested in sampling one instance of how GPT-4V would make a decision. Instead, estimating with what probability GPT-4V will choose this answer is more useful.

One way to estimate such probability robustly from samples with variance is through ensembling, a technique that has also been explored in other tasks [70]. Specifically, we propose to ensemble outputs from multiple slightly perturbed inputs. The key is to perturb input prompts to GPT-4V without changing the task or evaluation criteria. The input includes the text instruction, visual images, as well as the random seed. Our methods deploy different perturbations, including changing random seeds, the layout of renders, the number of rendered views, and the number of evaluation criteria. Figure 3 illustrates how we perturb the input and ensemble the results from these perturbed inputs together. Appendix D includes more details.

5.3. Quantifying Performance

We have now obtained a list of comparisons among a set of models $\mathcal{M}$. The comparisons are over a variety of sampled prompts denoted as $\mathcal{T}$ according to the user-defined criteria. Our goal is now to use this information to assign a number for each model in $\mathcal{M}$ such that it best explains the observed result. Our quantification method should consider the fact that the comparison results are samples from a probability distribution, as discussed in the previous subsection.

This problem is commonly studied in rating chess players, where a game between two players can have different outcomes even if one player is better than the other. In chess and many other competitions, the Elo score [18] is perhaps the most widely adapted method to produce a numerical estimation that reflects players' performance. The Elo rating system has also been adapted in prior works to evaluate image generative models [42, 60]. In this paper, we adapt the version proposed by Nichol et al. [42]. Specifically, let $\sigma_i \in \mathbb{R}$ denote the Elo score of the i^{th} model in $\mathcal{M}$. A higher score σ_i indicates better performance. We assume that the probability of model i beats model j is:

$$\Pr(\text{``}i \text{ beats } j\text{''}) = \left(1 + 10^{(\sigma_j - \sigma_i)/400}\right)^{-1}. \tag{1}$$

Our goal is to find score σ_i that can best explain the observed comparison results given the abovementioned assumption. This can be achieved via maximum likelihood estimation. Specifically, let A be a matrix where A_{ij} denotes the number of times model i beats model j in the list

Table 1. **Alignment with human judgment (higher is better).** Here we present Kendall's tau ranking correlation [30] between rankings provided by a metric and those provided by human experts. Higher correlation indicates better alignment with human judgment. We **bold-face** the most aligned method and underline the second place for each criterion. Our method achieves top-two performances for all evaluation criteria, while prior metrics usually only do well for at most two criteria.

Methods	Alignment	Plausibility	T-G Coherency	Tex Details	Geo Details	Average
PickScore [32]	0.667	0.484	0.458	0.510	0.588	0.562
CLIP-S [22]	0.718	0.282	0.487	0.641	0.667	0.568
CLIP-E [22]	0.813	0.426	**0.581**	0.529	0.658	0.628
Aesthetic-S [56]	0.795	0.410	0.564	0.769	0.744	0.671
Aesthetic-E [56]	0.684	0.297	0.555	0.813	0.684	0.611
Ours	**0.821**	**0.641**	0.564	**0.821**	**0.795**	**0.710**

Table 2. **Pairwise rating agreements (lower is better).** We measure the average probability that the decision of the metric matches that of human's for each comparison. Our method achieves strong alignment across most criteria.

Metrics	Align.	Plaus.	T-G.	Text.	Geo.	Avg.
PickS.	0.382	0.369	0.386	0.380	0.353	0.374
CLIP	0.384	0.441	0.423	0.375	0.374	0.400
Aest.	0.318	0.386	**0.353**	0.261	**0.311**	0.326
Ours	**0.292**	**0.278**	0.369	**0.244**	0.350	**0.307**

of comparisons. The final Elo score for this set of models can be obtained by optimizing the following objective:

$$\sigma = \arg\min_{\sigma} \sum_{i \neq j} A_{ij} \log\left(1 + 10^{(\sigma_j - \sigma_i)/400}\right). \quad (2)$$

In this paper, we initialize $\sigma_i = 1000$ and then use the Adam optimizer [31] to minimize the loss to obtain the final Elo score. Please refer to Appendix B.3 for more mathematical intuition about the formulation of the Elo score.

6. Results

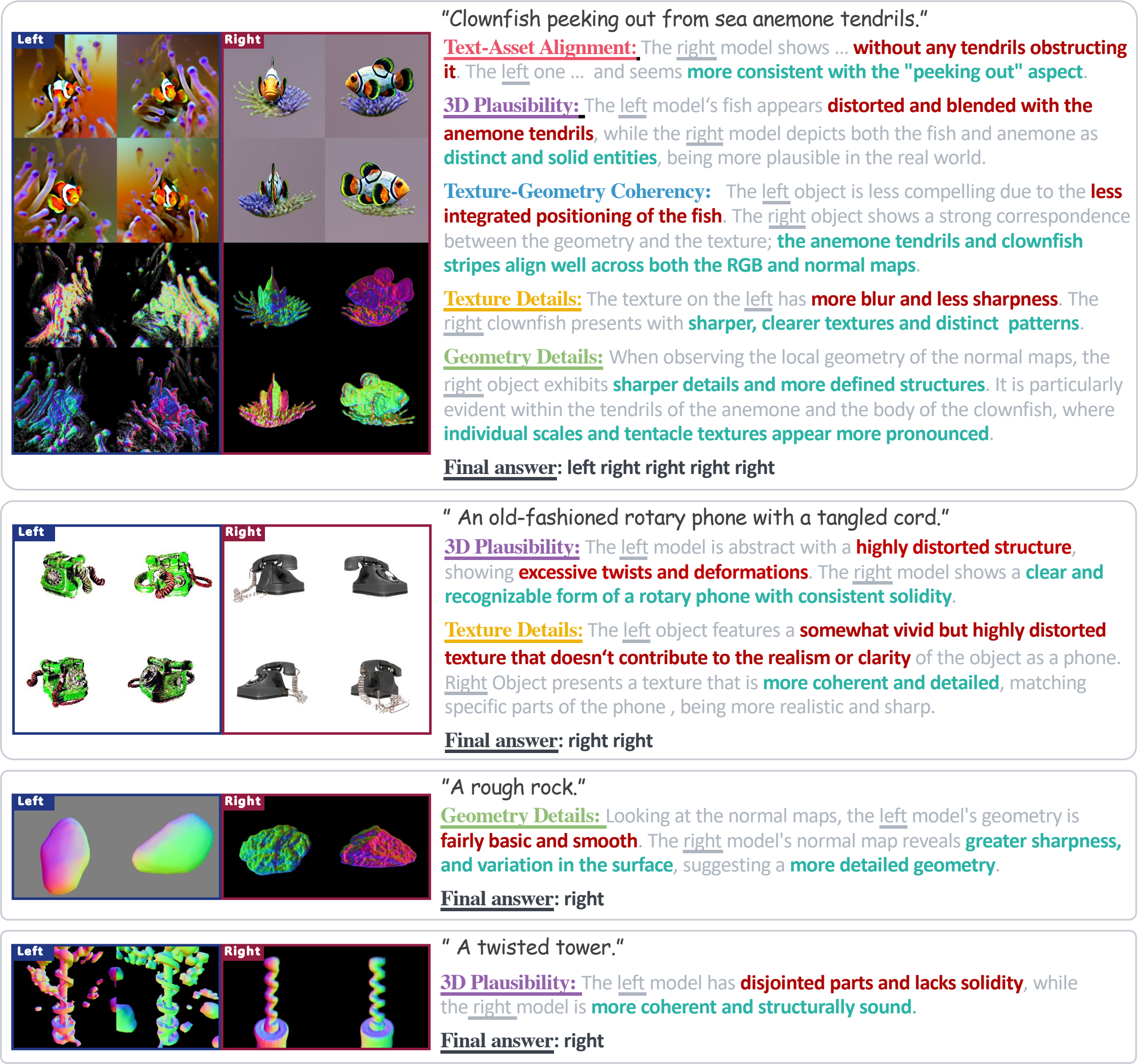
In this section, we provide a preliminary evaluation of our metric's alignment with human judgment across different criteria. We first introduce the experiment setup. We will discuss the main alignment results in Sec 6.1. Finally, we briefly showcase how to extend our models to different criteria in Sec 6.2. Analysis on more baseline methods and holistic evaluation can be find in the Appendix E.

Text-to-3D generative models to benchmark. We involve 13 generative models in the benchmark, including ten optimization-based methods and three recently proposed feed-forward methods. Please refer to Appendix C for the complete list. We leverage each method's official implementations when available. Alternatively, we turn to Threestudio's implementation [19]. For methods designed mainly for image-to-3D, we utilize Stable Diffusion XL [48] to generate images conditioned on text as input to these models. All experiments are conducted with default hyperparameters provided by the code.

Baselines metrics. We select three evaluation metrics with various considerations. **1) CLIP similarity** measures the cosine distance between the CLIP features [51] of the multi-view renderings and the text prompt. This metric is chosen because it is widely used in previous works as the metric for text–asset alignment [25, 28, 50]. **2) Aesthetic score** [56] is a linear estimator on top of CLIP that predicts the aesthetic quality of pictures. We choose this because it is trained on a large-scale dataset. **3) PickScore** [32] is a CLIP-based scoring function trained on the Pick-a-Pic dataset to predict human preferences over generated images. To compute the metrics above, we uniformly sample 30 RGB renderings for each of the generated assets. The CLIP similarity and aesthetic score can be directly computed from the multi-view renderings and averaged for each prompt. Since PickScore takes paired data as input for comparison, we assign 30 paired renderings for each pair of objects before averaging the PickScore results.

Evaluation criteria. While our method can potentially be applied to all user-defined criteria, in this work we focus on the following five criteria, which we believe are important for current text-to-3D evaluation tasks. **1) Text–asset alignment:** how well a 3D asset mirrors the input text description. **2) 3D plausibility:** whether the 3D asset is plausible in a real or virtual environment. A plausible 3D asset should not contain improbable parts such as multiple distorted faces (Janus problem) or noisy geometry floaters. **3) Texture details:** whether the textures and appearance of the shape are realistic, high resolution, and have appropriate saturation levels. **4) Geometry details:** whether the geometry makes sense and contains appropriate details. **5) Texture–geometry coherency:** whether geometry and textures agree with each other. For example, eyes of a character should be on reasonable parts of the face geometry.

Expert annotation. To evaluate the performance of our method, we need to conduct user preference studies to ob-



"Clownfish peeking out from sea anemone tendrils."

Text-Asset Alignment: The right model shows … **without any tendrils obstructing it**. The left one … and seems **more consistent with the "peeking out" aspect**.

3D Plausibility: The left model's fish appears **distorted and blended with the anemone tendrils**, while the right model depicts both the fish and anemone as **distinct and solid entities**, being more plausible in the real world.

Texture-Geometry Coherency: The left object is less compelling due to the **less integrated positioning of the fish**. The right object shows a strong correspondence between the geometry and the texture; **the anemone tendrils and clownfish stripes align well across both the RGB and normal maps**.

Texture Details: The texture on the left has **more blur and less sharpness**. The right clownfish presents with **sharper, clearer textures and distinct patterns**.

Geometry Details: When observing the local geometry of the normal maps, the right object exhibits **sharper details and more defined structures**. It is particularly evident within the tendrils of the anemone and the body of the clownfish, where **individual scales and tentacle textures appear more pronounced**.

Final answer: left right right right right

" An old-fashioned rotary phone with a tangled cord."

3D Plausibility: The left model is abstract with a **highly distorted structure**, showing **excessive twists and deformations**. The right model shows a **clear and recognizable form of a rotary phone with consistent solidity**.

Texture Details: The left object features a **somewhat vivid but highly distorted texture that doesn't contribute to the realism or clarity** of the object as a phone. Right Object presents a texture that is **more coherent and detailed**, matching specific parts of the phone , being more realistic and sharp.

Final answer: right right

"A rough rock."

Geometry Details: Looking at the normal maps, the left model's geometry is **fairly basic and smooth**. The right model's normal map reveals **greater sharpness, and variation in the surface**, suggesting a **more detailed geometry**.

Final answer: right

" A twisted tower."

3D Plausibility: The left model has **disjointed parts and lacks solidity**, while the right model is **more coherent and structurally sound**.

Final answer: right

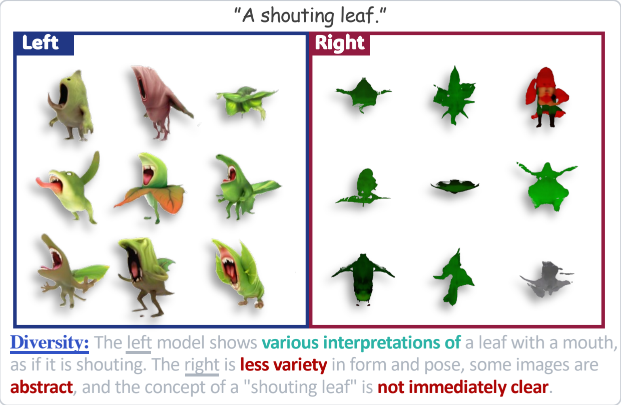
Figure 4. **Examples of the analysis by GPT-4V.** Given two 3D assets, we ask GPT-4V to compare them on various aspects and provide an explanation. We find that GPT-4V's preference closely aligns with that of humans.

tain ground truth preference data. Our user studies will present the input text prompt alongside a pair of 3D assets generated by different methods for the same input. The user will be asked to identify which 3D asset satisfies the criteria of interest better. We recruited 20 human experts who are graduate students experienced in computer vision and graphics research to annotate the data. We assigned 3 annotators per comparison question per evaluation criteria. We compute a reference Elo rating using the formula in Sec 5.3 using all expert annotations.

6.1. Alignment with Human Annotators.

In this section, we evaluate how well our proposed metric aligns with human preference. To achieve that, we use each metric to assign a score for each text-to-3D model for each evaluation criteria. Then, we compute Kendell's tau correlation [30] between the scores computed by the metrics and the reference scores. Table 1 shows the ranking correlations between scores predicted by different evaluation metrics and the reference Elo scores computed from expert annotators. We can see that our metrics achieve the best correlation in 4 out of 5 criteria, as well as the best average correlation. Note that our method achieves consistent performance across different criteria, while prior metrics usually perform well in only one or two. This highlights that our method is versatile in different evaluation criteria.

Our metric also shows strong human correlation for each 3D asset comparison question, which is a harder task. To measure that, we assume the response to each comparison question follows a Bernoulli distribution with probability p to select the first shape. Let p_i be the probability that the evaluation metric will select the first shape at question i and



Diversity: The left model shows various interpretations of a leaf with a mouth, as if it is shouting. The right is less variety in form and pose, some images are abstract, and the concept of a "shouting leaf" is not immediately clear.

Figure 5. **Diversity evaluation.** Our method can be extended to evaluate which text-to-3D models output more diverse 3D assets.

q_i be that of a human annotation. We measure the pairwise rating agreement using the L_1 distances: $\frac{1}{N}\sum_{i=1}^{N}|p_i - q_i|$, where N is the number of total questions. Table 2 shows that our method achieves top-two agreement across all but one criteria.

Figure 4 shows some exemplary outputs from our method. We can see that GPT-4V is also able to provide some analysis justifying its final choice.

6.2. Extension to Other Criteria

While we focus our empirical studies in five criteria, our metric can be adapted to evaluating a different criteria users might care about. For example, it is important that a generative model can produce different outputs given different random seeds. This aspect is commonly underlooked by most text-to-3D metrics. With small modification of the text and image prompt input into GPT-4V, our method can be applied to evaluate diversity. Figure 5 shows the visual image we provide GPT-4V when prompting it to answer the question about which model's output has more diversity. For each method, we produce 9 3D assets using different random seeds. We render each of these assets from a fixed camera angle to create the input image fed into GPT-4V. The text in Figure 5 is an excerpt of GPT-4V's answer. We can see that GPT-4V is able to provide a reasonable judgment about which image contains renders of more diverse 3D assets. Currently, we are restricted to qualitative studies because most existing text-to-3D models are still compute-intensive. We believe that large-scale quantitative study is soon possible with more compute-efficient text-to-3D models, such as Instant3D, becoming available.

7. Discussion

In this paper, we have presented a novel framework leveraging GPT-4V to establish a customizable, scalable, and human-aligned evaluation metric for text-to-3D generative tasks. First, we propose a prompt generator that can generate input prompts according to the evaluator's needs. Second, we prompt GPT-4V with an ensemble of customizable "3D-aware prompts." With these instructions, GPT-4V is able to compare two 3D assets according to an evaluator's need while remaining aligned to human judgment across various criteria. With these two components, we are able to rank text-to-3D models using the Elo system. Experimental results confirm that our approach can outperform existing metrics in various criteria.

Limitations and future work. While promising, our work still faces several unresolved challenges. First, due to limited resources, our experiment and user studies are done on a relatively small scale. It's important to scale up this study to better verify the hypothesis. Second, GPT-4V's responses are not always true. For example, GPT-4V sometimes shows hallucinations—a prevalent issue for many large pretrained models [72]. GPT-4V can also process some systematic errors, such as bias toward certain image positions [73, 74]. Such biases, if unknown, could induce errors in our evaluation metric. While our ensembling technique can mitigate these issues, how to solve them efficiently and fundamentally remains an interesting direction. Third, a good metric should be "un-gamable". However one could potentially construct adversarial patterns to attack GPT-4V. This way one might gain a high score without needing to produce high-quality 3D assets. Last, while our method is more scalable than conducting user preference studies, we can be limited by computation, such as GPT-4V API access limits. Our method also requires a quadratically growing number of comparisons, which might not scale well when evaluating a large number of models when compute is limited. It would be interesting to leverage GPT-4V to intelligently select input prompts to improve efficiency.

Acknowledgement. This project was in part supported by the National Key R&D Program of China (2022ZD0160201), Shanghai Artificial Intelligence Laboratory, Google, Samsung, Stanford HAI, Vannevar Bush Faculty Fellowship, ARL grant W911NF-21-2-0104, the Ministry of Education, Singapore, under its MOE AcRF Tier 2 (MOET2EP20221- 0012), NTU NAP, and under the RIE2020 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner(s). We would love to thank members of Stanford Computational Imaging Lab, Stanford Geometric Computation Group, Shanghai AI Lab, and Adobe Research for useful feedback and discussion.

References

[1] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas J. Guibas. Learning representations and generative models for 3d point clouds. In *International Conference on Machine Learning*, 2017. 2

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning. *ArXiv*, abs/2204.14198, 2022. 3

[3] Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Tachard Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Z. Chen, Eric Chu, J. Clark, Laurent El Shafey, Yanping Huang, Kathleen S. Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan A. Botha, James Bradbury, Siddhartha Brahma, Kevin Michael Brooks, Michele Catasta, Yongzhou Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, C Crépy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, M. C. D'iaz, Nan Du, Ethan Dyer, Vladimir Feinberg, Fan Feng, Vlad Fienber, Markus Freitag, Xavier García, Sebastian Gehrmann, Lucas González, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, An Ren Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wen Hao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Mu-Li Li, Wei Li, Yaguang Li, Jun Yu Li, Hyeontaek Lim, Han Lin, Zhong-Zhong Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussalem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alexandra Ros, Aurko Roy, Brennan Saeta, Rajkumar Samuel, Renee Marie Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniela Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Ke Xu, Yunhan Xu, Lin Wu Xue, Pengcheng Yin, Jiahui Yu, Qiaoling Zhang, Steven Zheng, Ce Zheng, Wei Zhou, Denny Zhou, Slav Petrov, and Yonghui Wu. Palm 2 technical report. *ArXiv*, abs/2305.10403, 2023. 3

[4] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Yitzhak Gadre, Shiori Sagawa, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. Openflamingo: An open-source framework for training large autoregressive vision-language models. *ArXiv*, abs/2308.01390, 2023. 3

[5] Sherwin Bahmani, Jeong Joon Park, Despoina Paschalidou, Xingguang Yan, Gordon Wetzstein, Leonidas J. Guibas, and Andrea Tagliasacchi. Cc3d: Layout-conditioned generation of compositional 3d scenes. *ArXiv*, abs/2303.12074, 2023. 2, 3

[6] Eslam Mohamed Bakr, Pengzhan Sun, Xiaogian Shen, Faizan Farooq Khan, Li Erran Li, and Mohamed Elhoseiny. Hrs-bench: Holistic, reliable and scalable benchmark for text-to-image models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20041–20053, 2023. 2

[7] Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Michal Podstawski, Hubert Niewiadomski, Piotr Nyczyk, et al. Graph of thoughts: Solving elaborate problems with large language models. *arXiv preprint arXiv:2308.09687*, 2023. 5

[8] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, T. J. Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *ArXiv*, abs/2005.14165, 2020. 3

[9] Eric Chan, Connor Z. Lin, Matthew Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J. Guibas, Jonathan Tremblay, S. Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3d generative adversarial networks. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16102–16112, 2021. 2

[10] 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. 1

[11] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. *arXiv preprint arXiv:2303.13873*, 2023. 2

[12] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam M. Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Benton C. Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier García, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat,

Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Díaz, Orhan Firat, Michele Catasta, Jason Wei, Kathleen S. Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways. *J. Mach. Learn. Res.*, 24:240:1–240:113, 2022. 3

[13] 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. *arXiv preprint arXiv:2212.08051*, 2022. 1

[14] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, Eli VanderBilt, Aniruddha Kembhavi, Carl Vondrick, Georgia Gkioxari, Kiana Ehsani, Ludwig Schmidt, and Ali Farhadi. Objaverse-xl: A universe of 10m+ 3d objects. *arXiv preprint arXiv:2307.05663*, 2023. 1

[15] 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. *arXiv preprint arXiv:2307.05663*, 2023. 2

[16] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 2

[17] Danny Driess, F. Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Ho Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Peter R. Florence. Palm-e: An embodied multimodal language model. In *International Conference on Machine Learning*, 2023. 3

[18] Arpad E Elo. The proposed uscf rating system, its development, theory, and applications. *Chess Life*, 22(8):242–247, 1967. 5

[19] Yuan-Chen Guo, Ying-Tian Liu, Ruizhi Shao, Christian Laforte, Vikram Voleti, Guan Luo, Chia-Hao Chen, Zi-Xin Zou, Chen Wang, Yan-Pei Cao, and Song-Hai Zhang. threestudio: A unified framework for 3d content generation, 2023. 6

[20] Zekun Hao, Arun Mallya, Serge J. Belongie, and Ming-Yu Liu. Gancraft: Unsupervised 3d neural rendering of minecraft worlds. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14052–14062, 2021. 2

[21] Yuze He, Yushi Bai, Matthieu Lin, Wang Zhao, Yubin Hu, Jenny Sheng, Ran Yi, Juanzi Li, and Yong jin Liu. T3bench: Benchmarking current progress in text-to-3d generation. *ArXiv*, abs/2310.02977, 2023. 2

[22] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*, 2021. 3, 6

[23] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1

[24] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and L. Sifre. Training compute-optimal large language models. *ArXiv*, abs/2203.15556, 2022. 3

[25] Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2room: Extracting textured 3d meshes from 2d text-to-image models. *ArXiv*, abs/2303.11989, 2023. 3, 6

[26] Yushi Hu, Hang Hua, Zhengyuan Yang, Weijia Shi, Noah A. Smith, and Jiebo Luo. Promptcap: Prompt-guided task-aware image captioning. *ArXiv*, abs/2211.09699, 2022. 3

[27] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, Kriti Aggarwal, Zewen Chi, Johan Bjorck, Vishrav Chaudhary, Subhojit Som, Xia Song, and Furu Wei. Language is not all you need: Aligning perception with language models. *ArXiv*, abs/2302.14045, 2023. 3

[28] Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 867–876, 2022. 2, 6

[29] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*, 2023. 2

[30] M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30:81–93, 1938. 6, 7

[31] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 6

[32] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *arXiv preprint arXiv:2305.01569*, 2023. 6

[33] Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, 2022. 2, 3

[34] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *ArXiv*, abs/2301.12597, 2023. 2, 3

[35] Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. *https://arxiv.org/abs/2311.06214*, 2023. 2

[36] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 300–309, 2023. 2, 3

[37] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Learning to generate multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023. 2

[38] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, and Wenping Wang. Wonder3d: Single image to 3d using cross-domain diffusion, 2023. 2

[39] David Lopez-Paz and Maxime Oquab. Revisiting classifier two-sample tests. *arXiv: Machine Learning*, 2016. 2

[40] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. *arXiv preprint arXiv:2211.07600*, 2022. 2

[41] 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. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2

[42] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 5

[43] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*, 2022. 2

[44] OpenAI. Gpt-4v(ision) system card. *OpenAI*, 2023. 3

[45] R OpenAI. Gpt-4 technical report. *arXiv*, pages 2303–08774, 2023. 2, 3

[46] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 1

[47] Ryan Po, Wang Yifan, and Vladislav Golyanik et al. State of the art on diffusion models for visual computing. In *arxiv*, 2023. 1, 2

[48] Dustin Podell, Zion English, Kyle Lacey, A. Blattmann, Tim Dockhorn, Jonas Muller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *ArXiv*, abs/2307.01952, 2023. 6

[49] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 2

[50] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv*, 2022. 1, 2, 6

[51] 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 *International Conference on Machine Learning*, 2021. 2, 6

[52] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan T. Barron, Yuanzhen Li, and Varun Jampani. Dreambooth3d: Subject-driven text-to-3d generation. *ArXiv*, abs/2303.13508, 2023. 3

[53] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 2

[54] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022.

[55] 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. 1, 2

[56] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. Laion-5b: An open large-scale dataset for training next generation image-text models. *ArXiv*, abs/2210.08402, 2022. 6

[57] Ho Youn Seo, Hayeon Kim, Gwanghyun Kim, and Se Young Chun. Ditto-nerf: Diffusion-based iterative text to omnidirectional 3d model. *ArXiv*, abs/2304.02827, 2023. 3

[58] Zhenwei Shao, Zhou Yu, Mei Wang, and Jun Yu. Prompting large language models with answer heuristics for knowledge-based visual question answering. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14974–14983, 2023. 3

[59] Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv:2308.16512*, 2023. 2

[60] Zhan Shi, Xu Zhou, Xipeng Qiu, and Xiaodan Zhu. Improving image captioning with better use of captions. *arXiv preprint arXiv:2006.11807*, 2020. 5

[61] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 1

[62] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *ArXiv*, abs/2309.16653, 2023. 2, 3

[63] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste

Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971, 2023. 3

[64] Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, S. M. Ali Eslami, Oriol Vinyals, Felix Hill, and Zacharias Janssen. Multimodal few-shot learning with frozen language models. In *Neural Information Processing Systems*, 2021. 3

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

[66] Zhenhailong Wang, Manling Li, Ruochen Xu, Luowei Zhou, Jie Lei, Xudong Lin, Shuohang Wang, Ziyi Yang, Chenguang Zhu, Derek Hoiem, Shih-Fu Chang, Mohit Bansal, and Heng Ji. Language models with image descriptors are strong few-shot video-language learners. *ArXiv*, abs/2205.10747, 2022. 3

[67] 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. *arXiv preprint arXiv:2305.16213*, 2023. 2

[68] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Huai hsin Chi, F. Xia, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903, 2022. 5

[69] Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge J. Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4540–4549, 2019. 2

[70] Zhengyuan Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. The dawn of lmms: Preliminary explorations with gpt-4v (ision). *arXiv preprint arXiv:2309.17421*, 9, 2023. 2, 3, 5

[71] Andy Zeng, Adrian S. Wong, Stefan Welker, Krzysztof Choromanski, Federico Tombari, Aveek Purohit, Michael S. Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, and Peter R. Florence. Socratic models: Composing zero-shot multimodal reasoning with language. *ArXiv*, abs/2204.00598, 2022. 3

[72] Muru Zhang, Ofir Press, Will Merrill, Alisa Liu, and Noah A. Smith. How language model hallucinations can snowball. *ArXiv*, abs/2305.13534, 2023. 8

[73] Xinlu Zhang, Yujie Lu, Weizhi Wang, An Yan, Jun Yan, Lianke Qin, Heng Wang, Xifeng Yan, William Yang Wang, and Linda Ruth Petzold. Gpt-4v (ision) as a generalist evaluator for vision-language tasks. *arXiv preprint arXiv:2311.01361*, 2023. 3, 8

[74] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*, 2023. 3, 8