Generate Any Scene: Synthetic Training and Evaluation Data for Generating Visual Content

Ziqi Gao[1,*], **Weikai Huang**[1,*], **Jieyu Zhang**[1], **Aniruddha Kembhavi**[2], **Ranjay Krishna**[1,2]
[1]University of Washington, [2]Allen Institute of Artificial Intelligence

Code: https://github.com/RAIVNLab/GenerateAnyScene

Dataset: GenerateAnyScene Dataset

Abstract

Recent advances in text-to-vision generation excel in visual fidelity but struggle with compositional generalization and semantic alignment. Existing datasets are noisy and weakly compositional, limiting models' understanding of complex scenes, while scalable solutions for dense, high-quality annotations remain a challenge. We introduce GENERATE ANY SCENE, a data engine that systematically enumerates scene graphs representing the combinatorial array of possible visual scenes. GENERATE ANY SCENE dynamically constructs scene graphs of varying complexity from a structured taxonomy of objects, attributes, and relations. Given a sampled scene graph, GENERATE ANY SCENE translates it into a caption for text-to-image or text-to-video generation; it also translates it into a set of visual question answers that allow automatic evaluation and reward modeling of semantic alignment. Using GENERATE ANY SCENE, we first design a self-improving framework where models iteratively enhance their performance using generated data. *SDv1.5* achieves an average **4%** improvement over baselines and surpassing fine-tuning on CC3M. Second, we also design a distillation algorithm to transfer specific strengths from proprietary models to their open-source counterparts. Using fewer than 800 synthetic captions, we fine-tune *SDv1.5* and achieve a **10%** increase in TIFA score on compositional and hard concept generation. Third, we create a reward model to align model generation with semantic accuracy at a low cost. Using GRPO algorithm, we fine-tune SimpleAR-0.5B-SFT and surpass CLIP-based methods by **+5%** on DPG-Bench. Finally, we apply these ideas to the downstream task of content moderation where we train models to identify challenging cases by learning from synthetic data.

1 Introduction

Despite the high-fidelity of modern generative models (text-to-image and text-to-video), we are yet to witness wide-spread adoption [1, 2, 3, 4, 5]. Controllability remains out of reach [6]. Generated content appears realistic but often falls short of semantic alignment [7, 8, 9, 10]. Users prompt models with a specific concept in mind. For example, when prompted to generate a scene of a "A black dog chasing after a rabbit that is eating the grass, in Van Gogh's style, with starlight lightening", some models are likely to generate an image of a dog but might miss the rabbit or get the style incorrect.

We hypothesize that these limitations stem not only from architectural bottlenecks but more fundamentally from the lack of structured, compositionally rich training data [3], especially those with uncommon compositions. Popular datasets such as LAION [11] and CC3M [12] predominantly consist of web-crawled image-caption pairs, which are inherently noisy, weakly compositional, and biased toward single-object, coarse-grained descriptions. Such datasets lack explicit grounding of object-attribute relations and multi-object interactions, restricting models' ability to generalize to

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

complex visual scenes. Efforts to enhance caption quality [3, 13] have demonstrated that enhancing the compositional density and semantic richness of captions can significantly improve generative performance. Nevertheless, manual curation of such dense compositional annotations is labor-intensive, while automatic annotation methods (e.g., via MLMs) suffer from hallucination and semantic noise.

Constructing a compositional dataset requires that we first define *the space of the visual content*. Scene graphs are one such representation of the visual space [14, 15, 16, 17, 18], grounded in cognitive science [19]. A scene graph represents objects in a scene as individual nodes in a graph. Each object is modified by attributes, which describe its properties. For example, attributes can describe the material, color, size, and location of the object in the scene. Finally, relationships are edges that connect the nodes. They define the spatial, functional, social, and interactions between objects [20]. For example, in a living room scene, a "table" node might have attributes like "wooden" or "rectangular" and be connected to a "lamp" node through a relation: "on top of". This systematic scene graph structure provides simple yet effective ways to define and model the scene. As such, scene graphs are an ideal foundation for systematically defining the compositional space of visual content in text-to-vision generation.

We introduce GENERATE ANY SCENE, a system capable of efficiently enumerating the space of scene graphs representing a wide range of visual scenes. GENERATE ANY SCENE composes scene graphs of any structure using a rich taxonomy of visual elements, translating each scene graph into an input caption and visual question answers to evaluate the output image or video. In particular, we first construct a rich taxonomy of visual concepts consisting of $28,787$ objects, $1,494$ attributes, $10,492$ relations, $2,193$ scene attributes from various sources. Based on these assets, GENERATE ANY SCENE can synthesize an almost infinite number of scene graphs of varying complexity [21]. Besides, GENERATE ANY SCENE allows configurable scene graph generation. For example, evaluators can specify the complexity level of the scene graph to be generated or provide a seed scene graph to be expanded. By automating these steps, our system ensures both scalability and adaptability, providing researchers and developers with diverse, richly detailed scene graphs and corresponding captions tailored to their specific needs. We also conduct comprehensive text-to-vision evaluations using our generated captions, as detailed in Appendix A.

We show that GENERATE ANY SCENE can allow generation models to self-improve. Our diverse captions can facilitate a framework to iteratively improve *Text-to-Vision generation* models using their own generations. Given a model, we generate multiple images, identify the highest-scoring one, and use it as new fine-tuning data to improve the model itself. We fine-tune *SDv1.5* [22] and achieve an average of **4%** performance boost compared with original models, and this method is even better than fine-tuning with the same amount of real images and captions from the Conceptual Captions CC3M over different benchmarks.

We also use GENERATE ANY SCENE to design targeted distillation algorithms. Using our evaluations, we identify limitations in open-sourced models that their proprietary counterparts excel at. Next, we distill these specific capabilities from proprietary models. For example, *DaLL-E 3* [3] excels particularly in generating composite images with multiple parts. We distill this capability into *SDv1.5*, effectively bridging the gap between *DaLL-E 3* and *SDv1.5*. After targeted fine-tuning, *SDv1.5* achieves a **10%** increase in TIFA score [23] for compositional tasks and hard concept generation.

Then we propose a low-cost scene graph-based reward model for RLHF [24] in text-to-image generation. By leveraging synthetic scene graphs generated by GENERATE ANY SCENE, we generate exhaustive question-answer pairs that cover all objects, attributes, and relationships in the caption. Our method enables fine-grained, compositional reward modeling without manual annotation or heavy LLM inference. With GRPO [25], we fine-tune SimpleAR-0.5B-SFT [26] using a scene graph reward model, achieving better compositional alignment than CLIP-based methods [27] (**+5%** on DPG-Bench [28]).

Finally, we apply GENERATE ANY SCENE to the downstream application of content moderation. Content moderation is a vital application, especially as *Text-to-Vision generation* models improve. A key challenge lies in the limited diversity of existing training data. To address this, we leverage GENERATE ANY SCENE to generate diverse and compositional captions, creating synthetic training data that complements existing datasets. By retraining a ViT-T [29] detector with our enriched dataset, we enhance its detection performance, particularly in cross-model and cross-dataset scenarios.

2 Generate Any Scene

In this section, we present GENERATE ANY SCENE (Figure 1), a data engine that systematically synthesizes diverse scene graphs in terms of both structure and content and translates them into corresponding captions.

Scene graph. A scene graph is a structured representation of a visual scene, where objects are represented as nodes, their attributes (such as color and shape) are properties of those nodes, and the relationships between objects (such as spatial or semantic connections) are represented as edges. In recent years, scene graphs have played a crucial role in visual understanding tasks, such as those found in Visual Genome [14] and GQA [30] for visual question answering (VQA). Their utility has expanded to various *Text-to-Vision generation* tasks. For example, the DSG [31] and DPG [10] benchmarks leverage scene graphs to evaluate how well generated images align with captions.

Taxonomy of visual elements. To construct a scene graph, we use three main metadata types: **objects**, **attributes**, and **relations**. We further introduce **scene attributes** that capture global visual contexts, such as art style, to facilitate comprehensive caption synthesis. The statistics and source of our metadata are shown in Table 1. Additionally, we build a hierarchical taxonomy that categorizes metadata into distinct levels and types, enabling fine-grained analysis. This structure supports precise content synthesis, from broad concepts like "flower" to fine-grained instances such as "daisy."

Table 1: Summary of the quantities and sources of visual elements.

Metadata Type	Number	Source
Objects	28,787	WordNet [32]
Attributes	1,494	Wikipedia [33], etc.
Relations	10,492	Synthetic Visual Genome [34]
Scene Attributes	2,193	Places365 [35], etc.

2.1 Generating data with scene graphs

Step 1: Scene graph structure enumeration and query. Our engine first generates and stores a variety of scene graph structures based on a specified level of **structural constraints**, such as complexity [36], average degree and the number of connected components. defined by the total number of objects, relationships, and attributes in each graph. The process begins by determining the number of object nodes, and then by systematically enumerating different combinations of relationships among these objects and their associated attributes. Once all graph structures satisfying the given constraints are enumerated, they are stored in a database for later use. This enumeration process is executed only once for each combination of structural parameters, allowing us to efficiently query the database for suitable templates when needed.

Step 2: Populate the scene graph structure with metadata. Given a generated scene graph structure, the next step involves populating the graph with metadata. For each object node, attribute node, and relation edge, we sample the corresponding content from our metadata. This process is highly customizable and controllable: users can define the topics and types of metadata to include, for instance, by selecting only commonsense metadata or specifying relationships between particular objects. By determining the scope of metadata sampling, we can precisely control the final content of the captions and easily extend the diversity and richness of scene graphs by adding new metadata.

Step 3: Sample scene attributes. We also include scene attributes that describe aspects such as the art style, viewpoint, time span (for video), and 3D attributes (for 3D content). These scene attributes are sampled directly from our metadata, creating a list that provides contextual details to enrich the description of the visual content.

Step 4: Translate scene graph to caption. We introduce an algorithm that converts scene graphs and a list of scene attributes into captions. The algorithm processes the scene graph in topological order, transforming each object, its attributes, and relational edges into descriptive text. To maintain coherence, it tracks each concept's occurrence, distinguishing objects with identical names using terms like "the first" or "the second." Objects that have been previously referenced without new

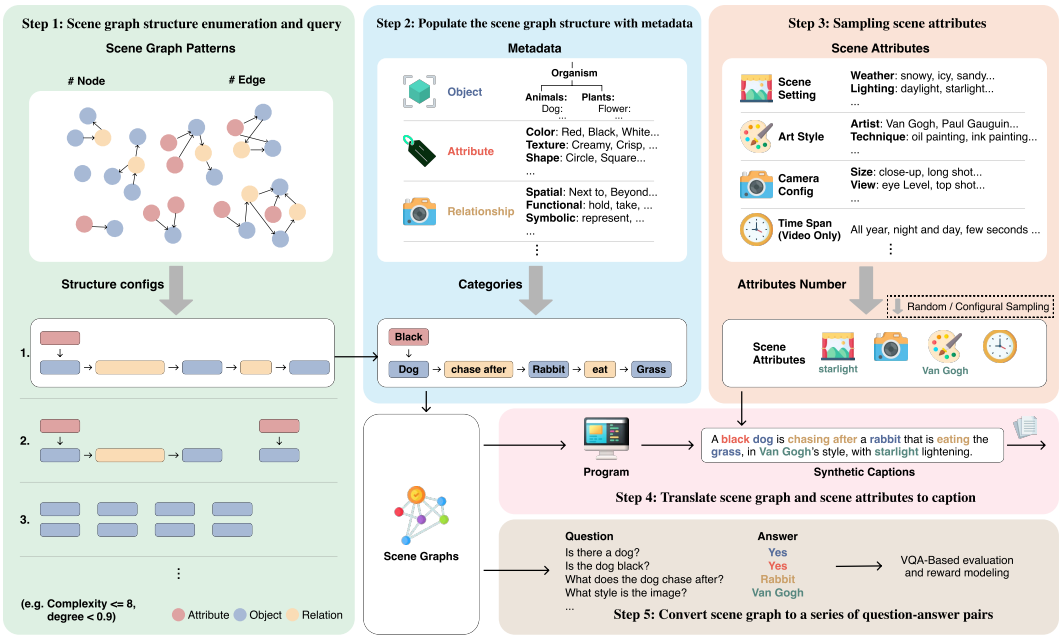


Figure 1: The generation pipeline of GENERATE ANY SCENE. **Step 1:** Enumerate diverse scene graph structures under user-defined constraints. **Step 2:** Populate structures with sampled objects, attributes, and relations. **Step 3:** Sample scene attributes such as style, perspective, or time span. **Step 4:** Translate scene graph and attributes into coherent captions. **Step 5:** Automatically generate QA pairs covering all elements for evaluation and reward modeling.

relations are skipped to avoid misreferencing. This approach enhances caption clarity by preventing repetition and maintaining a logical reference.

Step 5: Convert scene graph to a series of question-answer pairs. Given a synthetic scene graph, GENERATE ANY SCENE supports systematically enumerating exhaustive question-answer (QA) pairs that cover every compositional element. For instance, GENERATE ANY SCENE can generate questions about object attributes (e.g., What color is the sphere?), spatial relationships (e.g., What is to the left of the cube?), and so on, where each answer corresponds to a node (object or attribute) or an edge (relationship) in the scene graph. This method ensures comprehensive coverage of all objects, attributes, and relationships described in the caption, with negligible computational overhead. By automating this process, one can not only leverage VQA-based metrics [37, 31] to evaluate the generated images, but also construct a fine-grained, compositional reward model without requiring manual annotations or costly LLM inference.

3 Self-Improving models with synthetic captions

With GENERATE ANY SCENE, we develop a self-improvement framework to improve generative capabilities. By generating scalable compositional captions from scene graphs, GENERATE ANY SCENE expands the textual and visual space, allowing for a diversity of synthetic images that extend beyond real-world scenes. Our goal is to utilize these richly varied synthetic images to further boost model performance.

Iterative self-improving framework. Inspired by DreamSync [39], we designed an iterative self-improving framework using GENERATE ANY SCENE with *SDv1.5* as the baseline model. With *VQA Score*, which shows strong correlation with human evaluations on compositional images [37], we guide the model's improvement throughout the process. Specifically, GENERATE ANY SCENE generates 3×10K captions across three epochs. For each caption, *SDv1.5* generates 8 images, and the image with the highest *VQA Score* is selected. From each set of 10K optimal images, we then select the top 25% (2.5K image-caption pairs) as the training data for each epoch. In subsequent

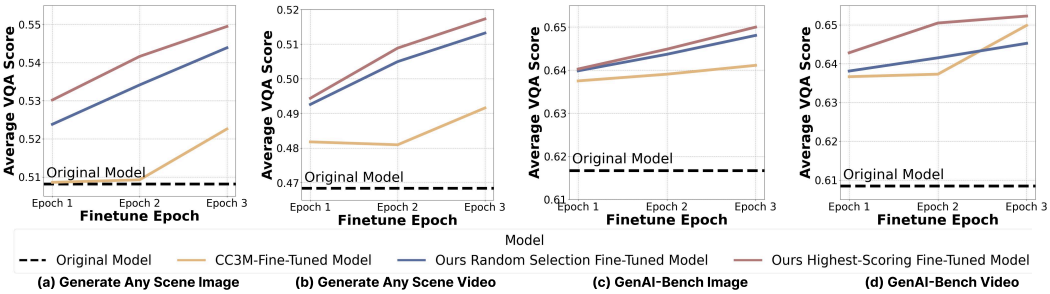


Figure 2: **Results for Self-Improving Models**. Average VQA score of *SDv1.5* fine-tuned on different data across 1K GENERATE ANY SCENE image/video evaluation set and GenAI-Bench image/video benchmark [38].

epochs, we use the fine-tuned model from the prior iteration to generate new images. We employ LoRA [40] for parameter-efficient fine-tuning.

Baselines. We conduct comparative experiments with the CC3M dataset, which comprises high-quality and diverse real-world image-caption pairs [12]. We randomly sample 3×10K captions from CC3M, applying the same top-score selection strategy for iterative fine-tuning of *SDv1.5*. Additionally, we include a baseline using random-sample fine-tuning strategy to validate the advantage of our highest-scoring selection-based strategy. We evaluate our self-improving pipeline on *Text-to-Vision generation* benchmarks, including GenAI Bench [38]. For the *Text-to-Video generation* task, we use *Text2Video-Zero* as the baseline model, substituting its backbone with the original *SDv1.5* and our fine-tuned *SDv1.5* models.

Fine-tuning with our synthetic captions can surpass high-quality real-world image-caption data. Our results show that fine-tuning with GENERATE ANY SCENE-generated synthetic data consistently outperforms CC3M-based fine-tuning across *Text-to-Vision generation* tasks (Figure 2), achieving the highest gains with our highest-scoring selection strategy. This highlights GENERATE ANY SCENE's scalability and compositional diversity, enabling models to effectively capture complex scene structures. Additional experiment settings and results are in Appendix C.

4 Distilling targeted capabilities

Although self-improving with GENERATE ANY SCENE shows clear advantages over high-quality real-world datasets, its efficiency is inherently limited by the model's own generation capabilities. To address this, we leverage the taxonomy and systematical generation capabilities within GENERATE ANY SCENE to identify specific strengths of proprietary models (*DaLL-E 3*), and distill these capabilities into open-source models. More details are in Appendix D.

We evaluate multiple models using GENERATE ANY SCENE controllably generated captions and observe that *DaLL-E 3* achieves *TIFA Score* **1.5** to **2** times higher than those of other models. As shown in Figure 4a, when comparing *TIFA Score* across captions with varying numbers of elements (objects, relations, and attributes), *DaLL-E 3* **counterintuitively** maintains consistent performance regardless of element count. The performance of other models declines as the element count increases, which aligns with expected compositional challenges. We suspect that these differences are primarily due to *DaLL-E 3*'s advanced capabilities in compositionality and **understanding hard concepts**, which ensures high faithfulness across diverse combinations of element types and counts.

Distilling compositionality from DaLL-E 3. When analyzing model outputs from our synthetic captions, we find that *DaLL-E 3* tends to produce straightforward combinations of multiple objects (Figure 3). In contrast, open-source models like *SDv1.5* often omit objects from the captions, despite being capable of generating each one individually. This difference suggests that *DaLL-E 3* may benefit from training data emphasizing multi-object presence, even without detailed layout or object interaction. Such training likely underpins *DaLL-E 3*'s stronger performance on metrics like *TIFA Score* and *VQA Score* that prioritize object inclusion. To effectively distill these compositional

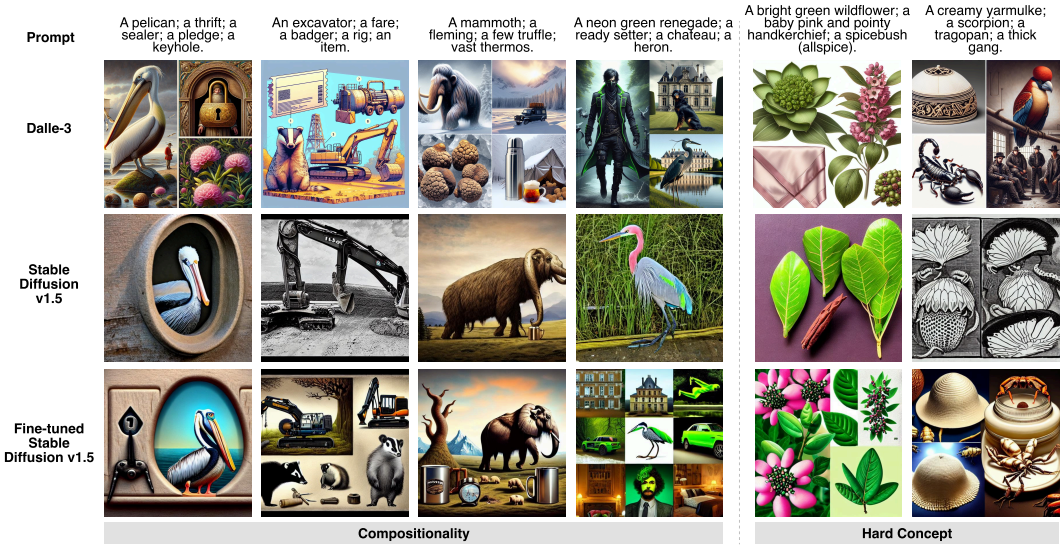
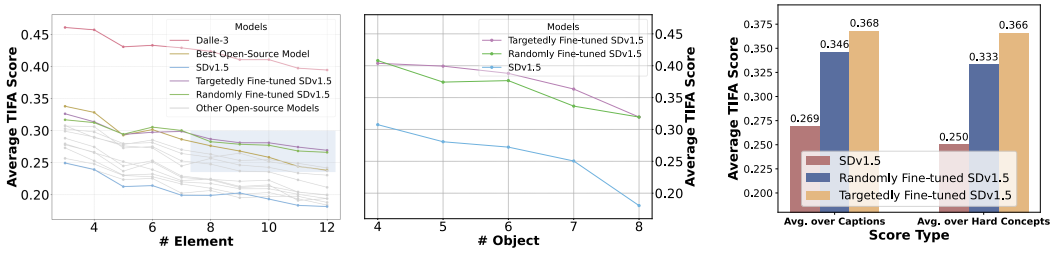


Figure 3: **Examples for Distilling Capabilities.** Examples of images generated by *DaLL-E 3*, the original *SDv1.5*, and the fine-tuned versions. The left four captions demonstrate fine-tuning with multi-object captions generated by GENERATE ANY SCENE for better compositionality, while the right two columns focus on understanding hard concepts.

abilities into *SDv1.5*, we employ GENERATE ANY SCENE for targeted synthesis of 778 multi-object captions, paired with images generated by *DaLL-E 3*, for finetuning *SDv1.5*.

Distilling hard concepts understanding from DaLL-E 3. Figure 3 shows that *DaLL-E 3* is capable not only of handling multi-object generation but also of understanding and generating rare and hard concepts, such as a specific species of flower. We attribute this to its training with proprietary real-world data. Using the taxonomy of GENERATE ANY SCENE, we compute model performance on each concept by averaging generation scores across captions containing that concept. Accumulating results through the taxonomy, we identify the 100 concepts where *SDv1.5* shows the largest performance gap relative to *DaLL-E 3*. For distilling, we generate 778 captions incorporating these hard concepts with other elements, and use *DaLL-E 3* to produce corresponding images.

Baselines. For the baseline, we randomly synthesize 778 captions using GENERATE ANY SCENE paired with *DaLL-E 3*-generated images to fine-tune the model. To evaluate model improvements, we generate another 1K multi-object captions and 1K hard-concept captions separately.



(a) **Distilling compositionality from DaLL-E 3**: Model results on TIFA vs. total element numbers in captions in 10K general GENERATE ANY SCENE captions.

(b) **Distilling compositionality from DaLL-E 3**: Model results on TIFA vs. total element numbers in captions in 1K multi-object GENERATE ANY SCENE captions.

(c) **Distilling hard concepts understanding from DALL-E 3**: Models' average *TIFA Score* performance over captions and hard concepts in 1K hard concepts GENERATE ANY SCENE captions.

Figure 4: **Results for Distilling Capabilities**. The left two figures show the results for **Distilling compositionality**, while the rightmost figure shows the results for **Distilling hard concepts understanding from DALL-E 3**.

Figure 5: **Comparison of generated images.** Our reward model enables image generation with better semantic alignment, realism, and visual quality than baselines.

Targeted caption synthesis via GENERATE ANY SCENE enables effective distillation of compositional abilities and hard concept understanding. We analyze images generated by *SDv1.5* before and after fine-tuning on high-complexity captions (Figure 3). Surprisingly, with fewer than 1K LoRA fine-tuning steps, *SDv1.5* effectively learns *DaLL-E 3* 's capability to arrange and compose multiple objects within a single image. Quantitatively, Figure 4b shows a 10% improvement in *TIFA Score* after targeted fine-tuning, surpassing the performance of the randomly fine-tuned model. On a broader set of 10K GENERATE ANY SCENE-generated captions, the targeted fine-tuned model consistently outperforms randomly fine-tuned and original counterparts across complex scenes (Figure 4a). These results confirm not only the effectiveness but also the scalability and efficiency of GENERATE ANY SCENE. Also, the results in Figure 4c show that our targeted fine-tuning with hard concepts leads to improved model performance, reflected in higher average scores across captions and increased scores for each challenging concept.

5 Reinforcement learning with a synthetic reward function

Reinforcement Learning with Human Feedback (RLHF) has become an increasingly popular fine-tuning strategy in text-to-image generation [41, 42, 26]. However, defining an effective reward model that accurately captures semantic alignment for text-to-image generation remains an open challenge. Existing reward models like CLIP offer only coarse-grained image-text similarity signals, which fall short in assessing compositional correctness and lack interpretability. Alternative approaches have explored using visual question answering (VQA) as a proxy for evaluating semantic alignment, aiming for finer-grained assessments, yet require either labor-intensive datasets with dense annotations or large volumes of contextually relevant questions via advanced LLMs. Leveraging its structured scene graph synthesis capabilities, GENERATE ANY SCENE offers a scalable alternative by producing exhaustive semantic queries with negligible overhead, enabling low-cost, compositional reward modeling (Sec 2.1).

Experiment setup. Building on this scene graph-based reward modeling strategy, we adopt Group Relative Policy Optimization (GRPO) as our reinforcement learning algorithm. We fine-tune the SimpleAR-0.5B-SFT model for one epoch using 10K captions generated by GENERATE ANY SCENE, each paired with their scene graph-derived QA sets. For reward evaluation, we use Qwen2.5-VL-3B, a lightweight open-source vision-language model, to answer these QA pairs given the model-generated images. The reward is computed as the accuracy across all questions. This fine-grained, scene graph-aligned reward provides precise feedback on compositional faithfulness. As a baseline, we compare against SimpleAR-0.5B-RL, trained with CLIP-based rewards on 11K captions from real world datasets for one epoch. We evaluate our scene graph-based reward model on three benchmarks: DPG-Bench [10], GenEval [9], and GenAI-Bench [38]. More details are in Appendix E.

GENERATE ANY SCENE rewards outperform CLIP. As shown in Table 2, our method outperforms both SFT and CLIP-RL models and achieves a significant improvement, demonstrating superior

compositional faithfulness driven by explicit scene graph rewards. Importantly, this performance gain is directly enabled by the GENERATE ANY SCENE engine, which constructs explicit scene graphs to generate compositional captions. GENERATE ANY SCENE provides a structured and cognitively aligned visual representation, from which we derive exhaustive QA pairs with minimal additional cost. Combined with lightweight VLM judge, this approach offers a scalable, low-cost solution for semantic-level reward modeling.

Table 2: Evaluation on the DPG, GenEval and GenAI benchmark. GRPO training with our reward model outperforms both SFT baseline and CLIP-RL models. TO: two objects, P: position, CA: color attribute.

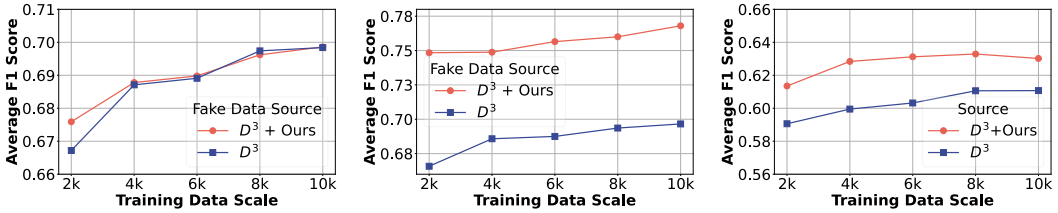
Method	DPG-Bench			GenEval				GenAI-Bench		
	Global	Relation	Overall	TO	P	CA	Overall	Basic	Advanced	All
SimpleAR-0.5B-SFT	85.02	86.59	78.48	0.73	0.22	0.23	0.53	0.74	0.60	0.66
SimpleAR-0.5B-RL (Clip)	86.64	88.51	79.66	**0.82**	0.26	**0.38**	0.59	**0.75**	0.60	0.67
SimpleAR-0.5B-RL (Ours)	**88.46**	**90.13**	**80.50**	0.81	**0.31**	**0.38**	**0.61**	**0.75**	**0.61**	**0.68**

6 Improving generated-content detection

Advances in *Text-to-Vision generation* underscore the need for effective content moderation [43]. Major challenges include the lack of high-quality and diverse datasets and the difficulty of generalizing detection across models *Text-to-Vision generation* [44, 45]. GENERATE ANY SCENE addresses these issues by enabling scalable, systematical generation of compositional captions, increasing the diversity and volume of synthetic data. This approach enhances existing datasets by compensating for their limited scope-from realistic to imaginative-and variability.

Experiment setup. To demonstrate GENERATE ANY SCENE's effectiveness in training generated content detectors, we used the D^3 dataset [46] as a baseline. We sampled 5K captioned real and SDv1.4-generated image pairs from D^3 and generated 5K additional images with GENERATE ANY SCENE captions. We trained a ViT–T [47] model with a single-layer linear classifier, and compared models trained with samples solely from D^3 against those trained with samples GENERATE ANY SCENE and D^3.

GENERATE ANY SCENE improves generated content detectors. We evaluate the detector's generalization on the GenImage [48] validation set and images generated using GENERATE ANY SCENE captions. Figure 6 demonstrates that combining GENERATE ANY SCENE-generated images with real-world captioned images consistently enhances detection performance, particularly across cross-model scenarios and diverse visual scenes. More details are in Appendix F.



(a) **In-domain testing (Same Model - SD v1.4)**: Detection results on images generated by SD v1.4 using the GenImage dataset.

(b) **In domain testing (cross-model)**:Average detection results on images generated by multiple models using our captions.

(c) **Out of domain**: Average detection results on images generated by multiple models using captions from the GenImage dataset.

Figure 6: **Results for Application 4: Generated content detector**. Comparison of detection performance across different data scales using D^3 alone versus the combined D^3 + GENERATE ANY SCENE training set in cross-model and cross-dataset scenarios.

7 Comprehensive evaluation with GENERATE ANY SCENE

We conduct extensive evaluations of text-to-vision models using GENERATE ANY SCENE. Specifically, we synthesize 10K captions for text-to-image, 10K for text-to-video, and 1K for text-to-3D,

covering diverse scene structures and content topics. We evaluate 12 text-to-image, 9 text-to-video, and 5 text-to-3D models. Evaluations combine GENERATE ANY SCENE synthetic scene graphs with existing metrics (e.g., CLIP Score [49], VQA Score [37], TIFA Score [23, 31]) to assess semantic similarity, faithfulness, and human preference alignment. Our key findings include: (1) DiT-backbone text-to-image models align more closely with input captions than UNet-backbone models. (2) Text-to-video models struggle with balancing dynamics and consistency, while both text-to-video and text-to-3D models show notable gaps in human preference alignment. Additionally, we leverage GENERATE ANY SCENE's controllable caption generation to conduct fine-grained evaluations. These analyses cover varying levels of perplexity, scene complexity, and commonsense reasoning, as well as performance across different content categories. Details are in Appendix A.

8 Related work

***Text-to-Vision generation* models.** *Text-to-Image generation* advances are driven by diffusion models and LLMs. Some open-source models [22, 50, 51, 52, 53, 54] use UNet backbones to refine images iteratively. In parallel, Diffusion Transformers (DiTs) architectures[55, 56, 57, 58] have emerged as a better alternative in capturing long-range dependencies and improving coherence. Proprietary models like DALL-E 3 [3] and Imagen 3 [59] still set the state-of-the-art. Based on *Text-to-Image generation* method, *Text-to-Video generation* models typically utilize time-aware architectures to ensure temporal coherence across frames [60, 61, 62, 63, 64, 65, 66, 67]. In *Text-to-3D generation*, recent proposed models [4, 68, 69, 70, 71] integrate the diffusion models with Neural Radiance Fields (NeRF) rendering to generate diverse 3D objects. Recent studies [26, 42, 72, 73] have also explored the integration of image generation into a unified multimodal language model (MLM) framework based on auto-regressive transformer architectures, demonstrating promising improvements in both performance and efficiency.

Synthetic captions for *Text-to-Vision generation*. Captions for *Text-to-Vision generation* models vary greatly in diversity, complexity, and compositionality. This variation makes it challenging and costly to collect large-scale and diverse captions written by humans. Consequently, synthetic captions have been widely used for both training [74, 39, 75, 76, 8, 77, 78, 79] and evaluation purposes [7]. For example, training methods like LLM-Grounded Diffusion [74] leverage LLM-generated captions to enhance the model's understanding and alignment with human instruction. For evaluation, benchmarks such as T2I-CompBench [7] and T2V-CompBench [8] utilize benchmarks generated by LLMs. However, LLMs are hard to control and may introduce exhibit systematic bias. In this work, we propose a programmatic scene graph-based data engine that can generate infinitely diverse captions for improving *Text-to-Vision generation* models.

Finetuning techniques for *Text-to-Vision generation*. To accommodate the diverse applications and personalization needs in text-to-vision models, numerous fine-tuning techniques have been developed. LoRA [40] reduces fine-tuning costs via low-rank weight updates, while Textual Inversion [80, 81] introduces new word embeddings for novel concepts without altering core parameters. DreamBooth [82] adapts models to specific subjects or styles using a few personalized images, and DreamSync [39] enables models to self-improve by learning from their own high-quality outputs. Recently, RLHF [26, 41, 42] in *Text-to-Vision generation* has shown promise as an efficient fine-tuning strategy. In this work, we use several fine-tuning techniques with GENERATE ANY SCENE to improve *Text-to-Vision generation* models.

9 Conclusion

We present GENERATE ANY SCENE, a system leveraging scene graph programming to generate diverse and compositional synthetic captions for *Text-to-Vision generation* tasks. It extends beyond existing real-world caption datasets to include comprehensive scenes and even implausible scenarios. To demonstrate the effectiveness of GENERATE ANY SCENE, we explore four applications: (1) self-improvement by iteratively optimizing models, (2) distillation of proprietary model strengths into open-source models, (3) a scene-graph-based efficient reward model within the GRPO, and (4) robust content moderation with diverse synthetic data. GENERATE ANY SCENE highlights the importance of synthetic data in improving *Text-to-Vision generation*, and addresses the need to systematically define and scalably produce the space of visual scenes.

References

[1] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators. 2024. *URL https://openai. com/research/video-generation-models-as-world-simulators*, 3, 2024.

[2] 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.

[3] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023.

[4] 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. *Advances in Neural Information Processing Systems*, 36, 2024.

[5] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James T. Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *ArXiv*, abs/2310.00426, 2023.

[6] Bohao Peng, Jian Wang, Yuechen Zhang, Wenbo Li, Ming-Chang Yang, and Jiaya Jia. Controlnext: Powerful and efficient control for image and video generation. *arXiv preprint arXiv:2408.06070*, 2024.

[7] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *ArXiv*, abs/2307.06350, 2023.

[8] Kaiyue Sun, Kaiyi Huang, Xian Liu, Yue Wu, Zihan Xu, Zhenguo Li, and Xihui Liu. T2v-compbench: A comprehensive benchmark for compositional text-to-video generation. *ArXiv*, abs/2407.14505, 2024.

[9] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36:52132–52152, 2023.

[10] Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. Ella: Equip diffusion models with llm for enhanced semantic alignment. *arXiv preprint arXiv:2403.05135*, 2024.

[11] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, 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.

[12] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018.

[13] Zejian Li, Chenye Meng, Yize Li, Ling Yang, Shengyuan Zhang, Jiarui Ma, Jiayi Li, Guang Yang, Changyuan Yang, Zhiyuan Yang, et al. Laion-sg: An enhanced large-scale dataset for training complex image-text models with structural annotations. *arXiv preprint arXiv:2412.08580*, 2024.

[14] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123:32–73, 2017.

[15] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3668–3678, 2015.

[16] Julian Ost, Fahim Mannan, Nils Thuerey, Julian Knodt, and Felix Heide. Neural scene graphs for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2856–2865, 2021.

[17] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1219–1228, 2018.

[18] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action genome: Actions as compositions of spatio-temporal scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10236–10247, 2020.

[19] Irving Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2):115, 1987.

[20] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 852–869. Springer, 2016.

[21] Jieyu Zhang, Weikai Huang, Zixian Ma, Oscar Michel, Dong He, Tanmay Gupta, Wei-Chiu Ma, Ali Farhadi, Aniruddha Kembhavi, and Ranjay Krishna. Task me anything. In *Advances in neural information processing systems*, 2024.

[22] 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 (CVPR)*, pages 10684–10695, June 2022.

[23] Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A Smith. Tifa: Accurate and interpretable text-to-image faithfulness evaluation with question answering, 2023.

[24] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155, 2022.

[25] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024.

[26] Junke Wang, Zhi Tian, Xun Wang, Xinyu Zhang, Weilin Huang, Zuxuan Wu, and Yu-Gang Jiang. Simplear: Pushing the frontier of autoregressive visual generation through pretraining, sft, and rl. *arXiv preprint arXiv:2504.11455*, 2025.

[27] 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, 2021.

[28] Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. Ella: Equip diffusion models with llm for enhanced semantic alignment, 2024.

[29] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[30] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019.

[31] Jaemin Cho, Yushi Hu, Roopal Garg, Peter Anderson, Ranjay Krishna, Jason Baldridge, Mohit Bansal, Jordi Pont-Tuset, and Su Wang. Davidsonian scene graph: Improving reliability in fine-grained evaluation for text-to-image generation. *ArXiv*, abs/2310.18235, 2023.

[32] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[33] Wikipedia Contributors. Lists of colors. `https://en.wikipedia.org/wiki/Lists_of_colors`, 2024. Accessed: 2024-11-09.

[34] Jae Sung Park, Zixian Ma, Linjie Li, Chenhao Zheng, Cheng-Yu Hsieh, Ximing Lu, Khyathi Chandu, Quan Kong, Norimasa Kobori, Ali Farhadi, Yejin Choi, and Ranjay Krishna. Synthetic visual genome. In *CVPR*, 2025.

[35] Alejandro López-Cifuentes, Marcos Escudero-Vinolo, Jesús Bescós, and Álvaro García-Martín. Semantic-aware scene recognition. *Pattern Recognition*, 102:107256, 2020.

[36] Madeleine Grunde-McLaughlin, Ranjay Krishna, and Maneesh Agrawala. Agqa: A benchmark for compositional spatio-temporal reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11287–11297, 2021.

[37] Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. *ArXiv*, abs/2404.01291, 2024.

[38] Baiqi Li, Zhiqiu Lin, Deepak Pathak, Jiayao Emily Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Genai-bench: A holistic benchmark for compositional text-to-visual generation. In *Synthetic Data for Computer Vision Workshop@ CVPR 2024*, 2024.

[39] Jiao Sun, Deqing Fu, Yushi Hu, Su Wang, Royi Rassin, Da-Cheng Juan, Dana Alon, Charles Herrmann, Sjoerd van Steenkiste, Ranjay Krishna, and Cyrus Rashtchian. Dreamsync: Aligning text-to-image generation with image understanding feedback. *ArXiv*, abs/2311.17946, 2023.

[40] J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685, 2021.

[41] Lixue Gong, Xiaoxia Hou, Fanshi Li, Liang Li, Xiaochen Lian, Fei Liu, Liyang Liu, Wei Liu, Wei Lu, Yichun Shi, et al. Seedream 2.0: A native chinese-english bilingual image generation foundation model. *arXiv preprint arXiv:2503.07703*, 2025.

[42] Dongzhi Jiang, Ziyu Guo, Renrui Zhang, Zhuofan Zong, Hao Li, Le Zhuo, Shilin Yan, Pheng-Ann Heng, and Hongsheng Li. T2i-r1: Reinforcing image generation with collaborative semantic-level and token-level cot. *arXiv preprint arXiv:2505.00703*, 2025.

[43] Gan Pei, Jiangning Zhang, Menghan Hu, Zhenyu Zhang, Chengjie Wang, Yunsheng Wu, Guangtao Zhai, Jian Yang, Chunhua Shen, and Dacheng Tao. Deepfake generation and detection: A benchmark and survey. *arXiv preprint arXiv:2403.17881*, 2024.

[44] Tianyi Wang, Xin Liao, Kam Pui Chow, Xiaodong Lin, and Yinglong Wang. Deepfake detection: A comprehensive survey from the reliability perspective. *ACM Computing Surveys*, 2024.

[45] Achhardeep Kaur, Azadeh Noori Hoshyar, Vidya Saikrishna, Selena Firmin, and Feng Xia. Deepfake video detection: challenges and opportunities. *Artificial Intelligence Review*, 57(6):1–47, 2024.

[46] Lorenzo Baraldi, Federico Cocchi, Marcella Cornia, Alessandro Nicolosi, and Rita Cucchiara. Contrasting deepfakes diffusion via contrastive learning and global-local similarities. *arXiv preprint arXiv:2407.20337*, 2024.

[47] Kan Wu, Jinnian Zhang, Houwen Peng, Mengchen Liu, Bin Xiao, Jianlong Fu, and Lu Yuan. Tinyvit: Fast pretraining distillation for small vision transformers. In *European conference on computer vision*, pages 68–85. Springer, 2022.

[48] Mingjian Zhu, Hanting Chen, Qiangyu Yan, Xudong Huang, Guanyu Lin, Wei Li, Zhijun Tu, Hailin Hu, Jie Hu, and Yunhe Wang. Genimage: A million-scale benchmark for detecting ai-generated image. *Advances in Neural Information Processing Systems*, 36, 2024.

[49] Tuhin Chakrabarty, Kanishk Singh, Arkadiy Saakyan, and Smaranda Muresan. Learning to follow object-centric image editing instructions faithfully. *ArXiv*, abs/2310.19145, 2023.

[50] 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.

[51] Daiqing Li, Aleks Kamko, Ehsan Akhgari, Ali Sabet, Linmiao Xu, and Suhail Doshi. Playground v2. 5: Three insights towards enhancing aesthetic quality in text-to-image generation. *arXiv preprint arXiv:2402.17245*, 2024.

[52] Pablo Pernias, Dominic Rampas, Mats L Richter, Christopher J Pal, and Marc Aubreville. Würstchen: An efficient architecture for large-scale text-to-image diffusion models. *arXiv preprint arXiv:2306.00637*, 2023.

[53] 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 (CVPR)*, pages 10684–10695, June 2022.

[54] DeepFloyd Lab at StabilityAI. DeepFloyd IF: a novel state-of-the-art open-source text-to-image model with a high degree of photorealism and language understanding. `https://www.deepfloyd.ai/deepfloyd-if`, 2023. Retrieved on 2023-11-08.

[55] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024.

[56] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-\alpha: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023.

[57] Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-\sigma: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. *arXiv preprint arXiv:2403.04692*, 2024.

[58] Black Forest Labs. Flux.1: Advanced text-to-image models, 2024. Accessed: 2024-11-10.

[59] Jason Baldridge, Jakob Bauer, Mukul Bhutani, Nicole Brichtova, Andrew Bunner, Kelvin Chan, Yichang Chen, Sander Dieleman, Yuqing Du, Zach Eaton-Rosen, et al. Imagen 3. *arXiv preprint arXiv:2408.07009*, 2024.

[60] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023.

[61] Fu-Yun Wang, Zhaoyang Huang, Xiaoyu Shi, Weikang Bian, Guanglu Song, Yu Liu, and Hongsheng Li. Animatelcm: Accelerating the animation of personalized diffusion models and adapters with decoupled consistency learning. *arXiv preprint arXiv:2402.00769*, 2024.

[62] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15954–15964, 2023.

[63] Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. Modelscope text-to-video technical report. *arXiv preprint arXiv:2308.06571*, 2023.

[64] Tianxing Wu, Chenyang Si, Yuming Jiang, Ziqi Huang, and Ziwei Liu. Freeinit: Bridging initialization gap in video diffusion models. In *European Conference on Computer Vision*, pages 378–394. Springer, 2025.

[65] Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7310–7320, 2024.

[66] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024.

[67] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all, March 2024.

[68] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

[69] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12619–12629, 2023.

[70] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12663–12673, 2023.

[71] 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*, pages 300–309, 2023.

[72] Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. *arXiv preprint arXiv:2410.13848*, 2024.

[73] Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025.

[74] Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. Llm-grounded diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models. *Trans. Mach. Learn. Res.*, 2024, 2023.

[75] Jialu Li, Jaemin Cho, Yi-Lin Sung, Jaehong Yoon, and Mohit Bansal. Selma: Learning and merging skill-specific text-to-image experts with auto-generated data. *ArXiv*, abs/2403.06952, 2024.

[76] Rui Zhao, Hangjie Yuan, Yujie Wei, Shiwei Zhang, Yuchao Gu, Lin Hao Ran, Xiang Wang, Zhangjie Wu, Junhao Zhang, Yingya Zhang, and Mike Zheng Shou. Evolvedirector: Approaching advanced text-to-image generation with large vision-language models, 2024.

[77] Dong Huk Park, Samaneh Azadi, Xihui Liu, Trevor Darrell, and Anna Rohrbach. Benchmark for compositional text-to-image synthesis. In *NeurIPS Datasets and Benchmarks*, 2021.

[78] Song Wen, Guian Fang, Renrui Zhang, Peng Gao, Hao Dong, and Dimitris Metaxas. Improving compositional text-to-image generation with large vision-language models. *ArXiv*, abs/2310.06311, 2023.

[79] Xindi Wu, Dingli Yu, Yangsibo Huang, Olga Russakovsky, and Sanjeev Arora. Conceptmix: A compositional image generation benchmark with controllable difficulty. *ArXiv*, abs/2408.14339, 2024.

[80] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6038–6047, 2022.

[81] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *ArXiv*, abs/2208.01618, 2022.

[82] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22500–22510, 2022.

[83] Spencer Sterling. zeroscope_v2_576w, 2023. Accessed: 2024-11-10.

[84] 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, 2023.

[85] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation, 2023.

[86] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024.

[87] Jiacheng Liu, Wenya Wang, Dianzhuo Wang, Noah A. Smith, Yejin Choi, and Hannaneh Hajishirzi. Vera: A general-purpose plausibility estimation model for commonsense statements, 2023.

[88] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.

[89] Y.C. Guo, Y.T. Liu, R. Shao, C. Laforte, V. Voleti, G. Luo, C.H. Chen, Z.X. Zou, C. Wang, Y.P. Cao, and S.H. Zhang. threestudio: A unified framework for 3d content generation. `https://github.com/threestudio-project/threestudio`, 2023.

[90] Giuseppe Vecchio and Valentin Deschaintre. Matsynth: A modern pbr materials dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22109–22118, 2024.

[91] Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. Material recognition in the wild with the materials in context database. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3479–3487, 2015.

[92] Jia Xue, Hang Zhang, Kristin Dana, and Ko Nishino. Differential angular imaging for material recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 764–773, 2017.

[93] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014.

[94] Zhe Xu, Dacheng Tao, Ya Zhang, Junjie Wu, and Ah Chung Tsoi. Architectural style classification using multinomial latent logistic regression. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*, pages 600–615. Springer, 2014.

[95] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

[96] Babak Saleh and Ahmed Elgammal. Large-scale classification of fine-art paintings: Learning the right metric on the right feature. *arXiv preprint arXiv:1505.00855*, 2015.

[97] Colby Crawford. 1000 cameras dataset. https://www.kaggle.com/datasets/crawford/1000-cameras-dataset, 2018. Accessed: 2024-11-09.

[98] Zijie J. Wang, Evan Montoya, David Munechika, Haoyang Yang, Benjamin Hoover, and Duen Horng Chau. DiffusionDB: A large-scale prompt gallery dataset for text-to-image generative models. *arXiv:2210.14896 [cs]*, 2022.

[99] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021.

[100] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- [redacted]: You should answer [Yes] , [No] , or [NA] .
  - [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
  - Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

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

   Answer: [Yes]

   Justification: Abstract and introduction accurately reflect the paper's contributions and scope

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: In the limitation section

Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [NA]

   Justification: [NA]

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: Our paper describes the pipeline in detail. We also open-sourced the code and the data for reproducing.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

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

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

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

   Answer: [Yes]

   Justification: We fully open-sourced our codebase and datasets as described in the beginning.

   Guidelines:

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

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Our paper specifies training settings, the dataset used, and the model across experiments. More details are in the Appendix.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: In experiment section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: In the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: [NA]

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: We discuss broader impacts in the Appendix and limitations.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: all the used assets are properly cited.

    Guidelines:

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

13. **New assets**

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

    Answer: [Yes]

    Justification: Please check the dataset host URL for documentation.

    Guidelines:

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

14. **Crowdsourcing and research with human subjects**

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

    Answer: [NA]

    Justification: [NA]

Guidelines:

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

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

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

    Answer: [NA]

    Justification: [NA]

    Guidelines:

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

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.