Scene Graph Disentanglement and Composition for Generalizable Complex Image Generation

Yunnan Wang[1,2] **Ziqiang Li**[1,2] **Wenyao Zhang**[1,2]
Zequn Zhang[2,3] **Baao Xie**[2] **Xihui Liu**[4] **Wenjun Zeng**[2] **Xin Jin**[2*]
[1]Shanghai Jiao Tong University, Shanghai, China
[2] Ningbo Institute of Digital Twin, Eastern Institute of Technology, Ningbo, China
[3]University of Science and Technology of China, Hefei, China
[4]The University of Hong Kong, Hong Kong, China
`{wangyunnan, ziqiangli}@sjtu.edu.cn` `jinxin@eitech.edu.cn`

Abstract

There has been exciting progress in generating images from natural language or layout conditions. However, these methods struggle to faithfully reproduce complex scenes due to the insufficient modeling of multiple objects and their relationships. To address this issue, we leverage the scene graph, a powerful structured representation, for complex image generation. Different from the previous works that directly use scene graphs for generation, we employ the generative capabilities of variational autoencoders and diffusion models in a generalizable manner, compositing diverse disentangled visual clues from scene graphs. Specifically, we first propose a Semantics-Layout Variational AutoEncoder (SL-VAE) to jointly derive *(layouts, semantics)* from the input scene graph, which allows a more diverse and reasonable generation in a one-to-many mapping. We then develop a Compositional Masked Attention (CMA) integrated with a diffusion model, incorporating *(layouts, semantics)* with fine-grained attributes as generation guidance. To further achieve graph manipulation while keeping the visual content consistent, we introduce a Multi-Layered Sampler (MLS) for an "isolated" image editing effect. Extensive experiments demonstrate that our method outperforms recent competitors based on text, layout, or scene graph, in terms of generation rationality and controllability. Code is available at `https://github.com/wangyunnan/DisCo`.

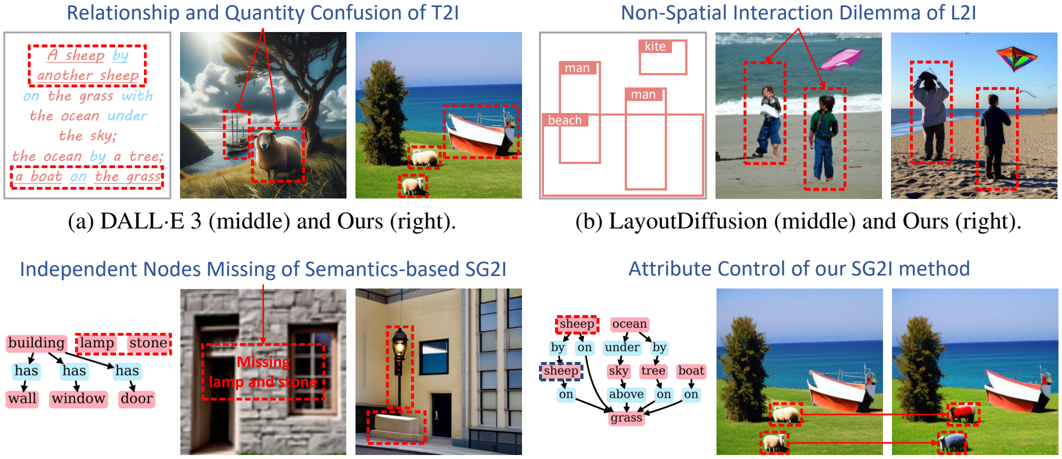
1 Introduction

Text-to-image (T2I) generation with diffusion models (DMs) [1, 2] has yielded remarkable advancements [3, 4, 5] in recent years, benefiting from the developments of vision-language foundation models [6, 7, 8, 9]. However, textual conditions with linear structure struggle to delineate the intricacies of complex scenes precisely. For example, as shown in the failure cases of DALL·E 3 [3] in Figure 1 (a), given the intricate text prompt "*A sheep by another sheep ... a boat on the grass.*", the T2I model may have difficulty accurately generating object relationships or quantities. Consequently, some studies [10, 11, 12, 13] strive to improve spatial relationship (e.g., "*by*" and "*on*") control by incorporating additional layout conditions. Nevertheless, as illustrated in the failure cases of LayoutDiffusion [10] in Figure 1 (b), layout-to-image (L2I) methods inevitably encounter challenges in representing certain non-spatial interactions, such as depicting "*playing*" within spatial topology.

To efficiently depict complex scenes for guiding generative models, recent methods [14, 15, 16] utilize structured scene graphs as conditions instead of text or layout prompts. Scene graphs [17]

[*] Xin Jin is the corresponding author.

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


Relationship and Quantity Confusion of T2I
A sheep by another sheep on the grass with the ocean under the sky; the ocean by a tree; a boat on the grass
Non-Spatial Interaction Dilemma of L2I
kite
man
man
beach
Independent Nodes Missing of Semantics-based SG2I
building
lamp
stone
has
has
has
wall
window
door
Missing lamp and stone
Attribute Control of our SG2I method
sheep
ocean
by
on
under
by
sheep
sky
tree
boat
on
above
on
on
grass

(a) DALL·E 3 (middle) and Ours (right).

(b) LayoutDiffusion (middle) and Ours (right).

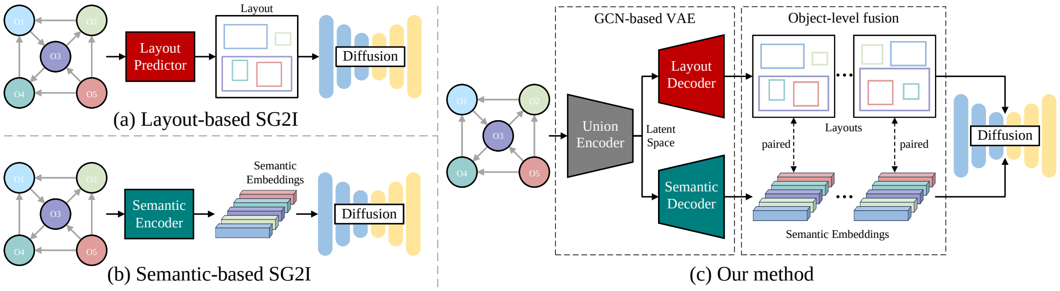
(c) R3CD (middle) and Ours (right).

(d) Results w/o (middle) and w/ (right) AC.

Figure 1: Failure cases generated by **(a)** text-to-image (T2I) (DALL·E 3 [3]), **(b)** layout-to-image (L2I) (LayoutDiffusion [10]), and **(c)** semantics-based scene-graph-to-image (SG2I) (R3CD [14]) methods . **(d)** Generalizable object *Attribute Control* (AC) under consistency achieved by our DisCo.

represent scenes with a structured graph format, where objects within the scene are denoted as nodes and the relationships between objects are represented as edges. Scene-Graph-to-Image (SG2I) generation is a challenging task due to the frequent ambiguous alignment between graph edges and relationships/interactions among visual objects. To address this issue, layout-based SG2I methods [15, 17, 18, 19] explicitly predict the spatial arrangements of objects in scenes by additional layout predictors, followed by L2I synthesis according to the layout (as demonstrated in Figure 2 (a)). These methods commonly employ one-to-one mapping, i.e., a single scene graph only corresponds to one layout, which severely limits the generation diversity. Besides, they also inherit the limitation of the L2I approach in modeling non-spatial interactions, whereby each object is typically generated independently. In contrast, as shown in Figure 2 (b), semantic-based SG2I methods implicitly encode graph edges into node embeddings by graph convolutional networks (GCNs), which effectively aligns object semantics with non-spatial interactions. Nonetheless, these methods are weak in logically determining the spatial positions of independent nodes, which might cause the absence of independent nodes (e.g., the "*lamp*" and "*stone*" shown in Figure 1 (c)) in the generated image.

In this paper, we propose **DisCo**, a **Co**mpositional image generation framework that integrates the **Dis**entangled layout and semantics derived from scene graph representations (as depicted in Figure 2 (c)). To boost the representational capacity of scene graphs for complex scenes, we augment the node and edge representations with CLIP [6] text embeddings, and incorporate extra spatial information (i.e., bounding box embeddings) for nodes during training. Once the textual scene graph is constructed, we propose a *Semantics-Layout Variational AutoEncoder* (SL-VAE) based on triplet-GCN [18] to jointly model the spatial relationships and non-spatial interactions in the scene. SL-VAE allows the one-to-many disentanglement for *spatial layout* and *interactive semantics* that match the



(a) Layout-based SG2I

(b) Semantic-based SG2I

(c) Our method

Figure 2: **Comparison between the previous SG2I architectures and ours**. **(a)** Layout-based SG2I model [15] generate a spatial arrangement with an object layout; **(b)** Semantic-based SG2I models [14, 16] build interactive semantic embedding between objects; **(c)** Our method leverages scene graph representation by jointly deriving the disentangled layout and semantics with the proposed SL-VAE.

input scene graph. This is achieved by sampling from a Gaussian distribution, offering object-level *(layouts, semantics)* conditions for the diffusion process [20]. Given that the layout and semantics encapsulate global and local relational information, we further introduce the *Compositional Masked Attention* (CMA) mechanism to inject object-level graph information with fine-grained attributes into the diffusion model, thereby preventing relational confusion and attribute leakage.Finally, we present a *Multi-Layered Sampler* (MLS) technique that leverages the diverse conditions generated by SL-VAE, achieving generalizable generation for object-level graph manipulation (i.e., node addition and attribute control) in the SG2I task, as depicted by the color change of two "*sheep*" in Figure 1 (d).

In summary, our key contributions are as follows: **(i)** We apply the textual scene graph as a structured scene representation and introduce the *Semantics-Layout Variational AutoEncoder* (SL-VAE) to disentangle diverse *spatial layouts* and *interactive semantics* from the scene graph; **(ii)** We present the *Compositional Masked Attention* (CMA) to inject extracted object-level graph information with fine-grained attributes into the diffusion model, which avoids relational confusion and attribute leakage; **(iii)** We introduce the *Multi-Layered Sampler* (MLS), a technique that leverages the diverse conditions produced by SL-VAE to implement object-level graph manipulation while keeping the visual content consistent; **(iv)** Our method outperforms current text/layout-based methods in relationship generation and achieves significantly superior generation performance compared to state-of-the-art SG2I models, thus showcasing the generalization of textual scene graphs in depicting complex scenes.

2 Preliminary

2.1 Text-to-Image Diffusion Models

Diffusion models (DMs) [1] are generative models that learn the data distribution $p(\boldsymbol{x})$ by gradually performing T-step noise reduction from the variables $\boldsymbol{x}_T$ sampled from the Gaussian distribution $\mathcal{N}(0,1)$. Thus the training process of DMs can be regarded as the reverse process of a Markov chain with a fixed length T. To generate high-resolution images with less computational resources, Latent Diffusion Models (LDMs) [20] encode the image $\boldsymbol{x}$ into the latent space $\boldsymbol{z}$ with the pre-trained Vector Quantized Variational AutoEncoder (VQ-VAE). Subsequently, the LDMs aim to predict the distribution $p(\boldsymbol{z})$ rather than $p(\boldsymbol{x})$. For the text-to-image LDMs, the text is encoded with the CLIP [6] text encoder E_{CLIP}. Then the objective function of a text-guided LDM can be formulated as follows:

$$\mathcal{L}_{LDM} = \mathbb{E}_{\boldsymbol{z},\boldsymbol{\epsilon}\sim\mathcal{N}(0,1),t}[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t, E_{\text{CLIP}}(\boldsymbol{c}), t)\|_2^2], \tag{1}$$

where $E_{\text{CLIP}}(\boldsymbol{c})$ is the text embedding of the text condition $\boldsymbol{c}$, t is the diffusion step, $\boldsymbol{\epsilon}_\theta$ is a model for estimating the noise $\boldsymbol{\epsilon}$, and $\boldsymbol{z}_t = \alpha_t \boldsymbol{z}_{t-1} + \sigma_t \boldsymbol{\epsilon}$ is the t-step noised latent code from the ground-truth $\boldsymbol{z}_0$. During inference, the model $\boldsymbol{\epsilon}_\theta$ with various samplers [1, 21] gradually denoises the initial noise $\boldsymbol{z}_T \sim \mathcal{N}(0,1)$. Finally, the predicted latent code is decoded into the image space.

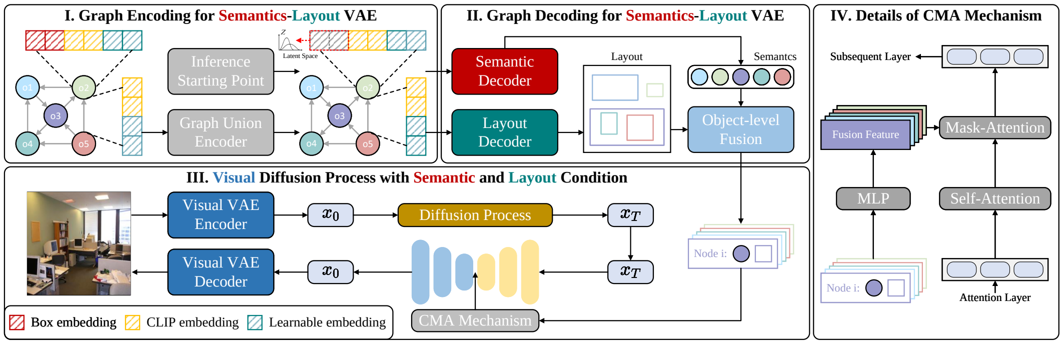


Figure 3: **Framework overview**. (I) We parameterize the node embeddings into the Gaussian distribution with the Graph Union Encoder, which jointly models the spatial relationships and non-spatial interactions in scene graphs; (II) The Semantic and Layout Decoders generate spatial layouts and interactive semantics sampled from Gaussian distribution, respectively; (III) A diffusion model with the proposed Compositional Masked Attention (CMA) incorporates object-level conditions to generate visual images following the scene graph description; (IV) Detailed structure of CMA Layer.

2.2 Scene Graph Representation

The scene graph $G = (O, E)$ [17] presents a structured scene representation. Nodes $O = \{o_i\}_{i=1}^{N_o}$ denotes N_o objects within the scene, while edges $E = \{e_{ij}\}_{1 \leq i,j \leq N_o, i \neq j}$ denotes relationships between objects. All nodes and edges come with a semantic label, denoted as $c_i^o \in \mathcal{C}^o$ and $c_{ij}^e \in \mathcal{C}^e$, where $\mathcal{C}^o$ and $\mathcal{C}^e$ are category vocabularies of nodes and edges, respectively. In practice, the nodes $O = \{o_i\}_{i=1}^{N_o}$ and the triples $T = \{t_{ij} = (o_i, e_{ij}, o_j)\}_{1 \leq i,j \leq N_o, i \neq j}$ representing connections from o_i to o_j serve as inputs for graph convolutional networks (GCNs). Moreover, nodes and edges are typically converted into learnable embeddings using embedding layers denoted as E_{emb}^o and E_{emb}^e.

3 Methodology

As illustrated in Figure 3, we present a novel SG2I synthesis framework known as DisCo. The DisCo comprises three primary components: (1) Semantics-Layout Variational AutoEncoder (SL-VAE) that disentangle diverse spatial layouts and interactive semantics from the scene graph (Section 3.1); (2) Compositional Masked Attention (CMA) that injects object-level *(layouts, semantics)* with fine-grained attributes into the diffusion model (Section 3.2); and (3) Multi-Layered Sampler (MLS) that implements generalizable generation for object-level graph manipulation (Section 3.3).

3.1 Semantics-Layout Variational AutoEncoder

Textual Scene Graph Construction. For constructing the scene graph representation, we employ the visual-language model to fully leverage the inherent disentangled semantics of the language, while simultaneously facilitating the alignment between images and scene graphs. Specifically, we augment the node and edge embeddings of the scene graph with CLIP [6] text embeddings. During training, we also incorporate spatial information for node embeddings by including bounding box coordinates (i.e., top-left corner and box size denoted as $b_i = (x_i, y_i, w_i, h_i)$). Then the node embeddings $\mathcal{O}$ and edge embeddings $\mathcal{E}$ can be formulated as follows:

$$\mathcal{O} = \{E_{emb}^o(c_i^o) \otimes E_{\text{CLIP}}(o_i) \otimes E_{box}(b_i)\}_{i=1}^{N_o}, \ \mathcal{E} = \{E_{emb}^e(c_{ij}^e) \otimes E_{\text{CLIP}}(t_{ij})\}_{1 \leq i,j \leq N_o, i \neq j} \quad (2)$$

where E_{CLIP} denotes the frozen pre-trained text encoder, E_{box} is the spatial encoder for bounding box coordinates using Multi-Layer Perceptions (MLPs), and $\otimes$ denotes concatenate operation.

Graph Union Encoding. Although layout-based SG2I methods are superior in modeling spatial topology compared to the semantics-based method, they fall short in capturing object interactions (i.e., non-spatial relationships) within the scene. Accordingly, after obtaining the node and edge embeddings mentioned above, we apply a Conditional Variational Autoencoder (CVAE) [22] based on triplet-GCN [18] to jointly model the layout and semantics information. As shown in Figure 3.I, the L-layer Graph Union Encoder E_u takes node and edge embeddings as inputs:

$$(\phi_i^{l+1}, \phi_{ij}^{l+1}, \phi_j^{l+1}) = \text{GCN}_l(\phi_i^l, \phi_{ij}^l, \phi_j^l), l \in \{0, \dots, L-1\} \quad (3)$$

where l denotes the layer index of Graph Union Encoder, and ϕ denotes intermediate features. Here we initialize $(\phi_i^0, \phi_{ij}^0, \phi_j^0) = (\mathcal{O}_i, \mathcal{E}_{ij}, \mathcal{O}_j)$. Please refer to the **Appendix** for more details about the triplet-GCN. Given that the last node embedding ϕ_i^L integrates both topology and interaction information, we conduct layout-semantic modeling by parameterizing it into Gaussian spaces $Z \sim \mathcal{N}(\mu, \sigma)$. In this context, the means $\mu \in \mathbb{R}^{D_z}$ and variances $\sigma \in \mathbb{R}^{D_z}$ are estimated individually by two supplementary MLPs, where D_z denotes the dimensional of latent space for node embedding. Hence, we jointly model the layout and semantics through the following minimization:

$$\mathcal{L}_{union} = \text{KL}\left(E_u(u|y, \mathcal{O}, \mathcal{E}) \parallel p(u|y)\right), \quad (4)$$

where KL denotes the Kullback-Liebler divergence, y denotes condition and the prior $p(u|y)$ is the standard Gaussian distribution $\mathcal{N}(u \mid 0, 1)$. Specifically, we condition the latent space of the graph structure using the edge embedding following Equation 2 alongside the updated node embedding $\{E_{emb}^o(c_i^o) \otimes E_{\text{CLIP}}(o_i) \otimes u_i\}_{i=1}^{N_o}$, where u_i is a random vector sampled from Z. This architecture ensures that layout is solely necessary for training, with no need for hand-crafted layout in inference.

Disentangled Semantics-Layout Decoding. As illustrated in Figure 3.II, we disentangle the explicit *spatial layout* and implicit *interactive semantics* from the latent space using two separate triplet-GCN-based decoders, i.e., layout decoder D_l and semantic decoder D_s. The proposed Semantics-Layout

Variational AutoEncode (SL-VAE) comprises these two decoders and the graph union encoder mentioned above, which derives the spatial topology and object interactions from the scene graph representation. During training, the layout decoder is optimized by the following objective function:

$$\mathcal{L}_{layout} = \frac{1}{N_o} \sum_{i=1}^{N_o} |b_i - \hat{b}_i|_1, \tag{5}$$

where $\hat{b}_i$ denotes the predicted coordinates. We only incorporate the ground truth layout $\mathcal{B} = \{b_i\}_{i=1}^{N_o}$ during training, while generating N_l diverse *layouts* $\{\hat{\mathcal{B}}_n = \{\hat{b}_{n,i}\}_{i=1}^{N_o}\}_{n=1}^{N_l}$ by sampling Gaussian noise at inference time. For simplicity, we omit the superscript ˆ in the following description. The semantic decoder D_s generates *semantics* embeddings $\mathcal{S} = \{s_i\}_{i=1}^{N_o}$ to facilitate subsequent diffusion processes, and its parameters are iteratively updated with the diffusion loss in the next Section 3.2.

3.2 Diffusion with Compositional Masked Attention

Object-level Fusion Tokenizer. We integrate the *spatial layout* $\mathcal{B} = \{b_i\}_{i=1}^{N_o}$ and *interactive semantics* $\mathcal{S} = \{s_i\}_{i=1}^{N_o}$ at object level, as illustrated in Figure 3.III. The single-object embeddings $\mathcal{C} = \{c_i\}_{i=1}^{N_o} = \{s_i \otimes \mathcal{F}(b_i)\}_{i=1}^{N_o}$ are acquired by directly applying semantic embeddings, while encoding box information using a Fourier mapping $\mathcal{F}$ [23]. We define a learnable null embedding to pad the embedding length to N_{max}, thereby accommodating varying numbers of objects:

$$c_i = \begin{cases} s_i \otimes \mathcal{F}(b_i), & i \leq N_o \\ c_{null}, & \text{otherwise} \end{cases} \tag{6}$$

where c_{null} denotes the learnable null embedding for padding. We optionally add attribute embedding $\mathcal{A} = \{a_i\}_{i=1}^{N_{max}}$ to construct updated $\mathcal{C} = \{c_i \otimes a_i\}_{i=1}^{N_{max}}$, where c_i and a_i are separately processed by two MLPs before concatenation. Note that a_i is obtained similarly to edge embedding in Equation 2. We also define a learnable null embedding a_{null} for cases where no attribute is specified.

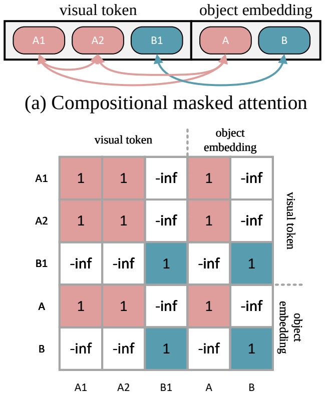
Compositional Masked Attention. The cross-attention mechanism in diffusion bridges the visual and textual information, while self-attention captures self-related information within visual tokens [24]. Therefore, we insert our proposed Compositional Masked Attention (CMA) between self-attention and cross-attention layers. This technique effectively injects graph information into the diffusion process at the object level, preventing semantic confusion and attribute leakage through the attention mask. Specifically, we denote the visual token output by the vanilla self-attention as $\mathcal{V} \in \mathbb{R}^{N_v \times D_v}$, where N_v and D_v represent the number and dimensions of tokens, respectively. Then the CMA layer can be expressed as:

$$\hat{\mathcal{V}} = SA_{mask}(\mathcal{V} \otimes \hat{\mathcal{C}}, \mathbf{M})[: N_v], \tag{7}$$

where $\hat{\mathcal{C}} = \{\hat{c}_i\}_{i=1}^{N_{max}}$ denotes object embeddings whose dimensions are aligned with the visual token $\mathcal{V}$ using MLPs. The matrix $\mathbf{M} \in \mathbb{R}^{(N_v+N_{max})\times(N_v+N_{max})}$ denotes the attention mask that depends on layout $\mathcal{B}$, which can be constructed as follows:

$$\mathbf{M}_{i,j} = \begin{cases} 1, & \text{if } i, j \text{ fall into the same object} \\ -inf, & \text{otherwise} \end{cases} \tag{8}$$

where "i, j fall into the same object" means that i and j index the visual tokens or object embeddings of the same object. Figure 4 illustrates the mechanism of CMA through a toy example. In contrast to vanilla self-attention, the proposed CMA prevents relational confusion and attribute leakage between different objects through the well-designed object-level masks mentioned above. As shown in Figure 3.IV, we forward the output of the CMA layer into the subsequent layers, serving as the updated visual token.



(a) Compositional masked attention

	A1	A2	B1	A	B
A1	1	1	-inf	1	-inf
A2	1	1	-inf	1	-inf
B1	-inf	-inf	1	-inf	1
A	1	1	-inf	1	-inf
B	-inf	-inf	1	-inf	1

(b) Attention mask

Figure 4: **Toy example** of (a) compositional masked attention, and (b) its corresponding attention mask. We use visual tokens and object embeddings of objects A and B for demonstration. A and B have 1 and 2 visual tokens, respectively, whose attribution is determined by bounding boxes.

Diffusion Loss. Based on the object semantics output from the proposed SL-VAE, we optimize the evidence lower bound between sampling noise and prediction noise conditioned on object-level

Table 1: **Performance comparison** on COCO-Stuff and Visual Genome datasets using Inception Score (IS) and Fréchet Inception Distance (FID) metrics. We report the results of methods with two generator structures, namely GAN- and Diffusion-based. The architecture of these methods is based on the layout (L) or semantics (S), while our approach includes both. The best results are **bolded**.

Method		Type	COCO [26]		Visual Genome [27]	
			IS ↑	FID ↓	IS ↑	FID ↓
Real Image		-	30.7	-	27.3	-
GAN-based	SG2Im [17]	L	8.2	99.1	7.9	90.5
	PasteGAN [28]	L	12.3	79.1	8.1	66.5
	SOAP [18]	L	14.5	81.0	-	-
	WSGC [19]	L	6.5	121.7	9.8	84.1
	KCGM [29]	S	-	-	11.6	27.4
Diffusion-based	LDM [20]	S	22.2	63.8	16.5	45.7
	SGDiff [16]	S	17.8	36.2	16.4	26.0
	SceneGenie [15]	L	22.2	63.3	20.3	42.2
	R3CD [14]	S	19.5	32.9	18.9	23.4
	DisCo (ours)	L+S	**23.1**	**30.8**	**22.3**	**21.9**

information (i.e., ground truth spatial layout and generated interactive semantics). Then the training loss of the diffusion model equipped with CMA can be summarized as follows:

$$\mathcal{L}_{LDM} = \mathbb{E}_{\boldsymbol{z},\boldsymbol{\epsilon}\sim\mathcal{N}(0,1),t}[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t, E_{\text{CLIP}}(O), \mathcal{B}, \mathcal{S}, \mathcal{A}, t)\|_2^2]. \tag{9}$$

Finally, we employ an end-to-end joint training pipeline for the whole proposed DisCo framework. The total objective function is presented as follows:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{LDM} + \lambda_2 \mathcal{L}_{union} + \lambda_3 \mathcal{L}_{layout}, \tag{10}$$

where λ_1, λ_2, and λ_3 are hyperparameters, which are typically set to 1.0, 0.1 and 1.0, respectively.

3.3 Multi-Layered Sampler

Manipulations in the input scene graph, such as node addition and attribute adjustment, pose challenges for maintaining visual consistency in the generated images, ultimately compromising generalizability. To achieve an "isolated" image editing effect, we also provide the Multi-Layered Sampler (MLS) motivated by SceneDiffusion [25]. The scheme defines each object as a layer, thus allowing independent object-level Gaussian sampling. In contrast to SceneDiffusion which scrambles the reference layouts randomly, we sample additional N_l (layouts, semantics) by the SL-VAE. Note that N_l fixed seeds exist for the same scene. Then we aggregate latent codes from various layers into $\boldsymbol{z}_n$ and utilize layout-converted non-overlapping masks $\{\mathcal{M}_n = \{m_{n,i}\}_{i=1}^{N_o}\}_{n=1}^{N_l}$ for locally conditioned diffusion. During the inference, the noise estimation for N_l scenes is calculated as follows:

$$\hat{\boldsymbol{\epsilon}}_n^{(t)} = \sum_{i=1}^{N_o} m_{n,i} \odot \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_n^{(t)}, E_{\text{CLIP}}(o_i), b_{n,i}, s_{n,i}, a_i, t), \tag{11}$$

Subsequently, the latent code for each object is computed as the weighted average of the N_l cropped denoised views. Please refer to the **Appendix** for more details about MLS.

4 Experiments

Dataset. We conduct scene-graph-to-image (SG2I) generation experiments on the Visual Genome (VG) [27] and COCO-Stuff (COCO) [26] datasets. The VG dataset comprises $108,077$ image-scene graph pairs, accompanied by the bounding box coordinates and object attributes. Following previous work [17], we select objects and relationships that appear at least $2,000$ and 500 times respectively in VG, resulting in 178 objects and 45 unique relationship types. Also, we ignore small objects and use images containing 5 to 30 objects along with a minimum of 3 relationships. Based on the above filtering, we have $62,565$ images available for training, each containing an average of 10 objects and 5 relationships. While the original COCO-Stuff dataset [26] lacks scene graph annotations, it consists of $40,000$ images annotated with bounding box coordinates and captions, essential for synthesizing geometric scene graphs [15, 16]. All images in the COCO-Stuff dataset are labeled as 80 item categories and 91 stuff categories.

Implementation Details. We fine-tune the pre-trained Stable-Diffusion 1.5[1] with the modified Attention module on 4 NVIDIA A100 GPUs, each with 80GB of memory. We apply the CLIP text encoder (vit-large-patch14) to construct the textual scene graph. We train the model with a batch size of 64 using the AdamW optimizer [30] with an initial learning rate of 1.0×10^{-4}, which is adjusted linearly over $50,000$ steps. During inference, we use the 50-step PNDMScheduler [21] with a classifiers-free scale [31] of 7.5. The sample number N_l in the multi-layered sampler is set to 5.

Evaluation Metrics. Following previous works [14, 15, 16], we evaluate the performance of our method with the Inception Score (IS) [32] and the Fréchet Inception Distance (FID) [33]. The IS score is derived from a pre-trained Inception Net [34], assessing both the quality and diversity of synthesized images. The FID score quantifies the dissimilarity between the generated image and the real image distribution, which evaluates the fidelity of the generated images. To measure the effectiveness of compositional generation, we further evaluate our method on the T2I-CompBench [35]. Besides, we apply CLIP for zero-shot attribute classification of the controlled object cropped by the bounding box, and subsequently evaluate the attribute control performance by the classification accuracy ACC$_{attr}$.

Quantitative Comparisons. To demonstrate the effectiveness of the proposed DisCo, we compare it with current state-of-the-art SG2I methods on the COCO-Stuff and Visual Genome datasets, which are summarized in Table 1. Our DisCo outperforms other methods in both IS and FID scores, revealing its superior performance in both fidelity and diversity of image generation. Compared with previous methods, the primary architectural advantage of DisCo is its innovative approach of simultaneously integrating disentangled layout and semantics extracted from scene graph representations. Moreover, the proposed SL-VAE achieves the diverse generation of layouts and semantics from a single scene graph through Gaussian distribution sampling. Therefore, our DisCo integrates the benefits of both layout-based and semantic-based methods, which is further ablated in detail in Table 4 of the ablation study. We proceed to assess the compositional generation on the T2I-CompBench [35], as shown in Table 2. The benchmark evaluates the competency of the text-to-image model in responding to compositional prompts. We report UniDet, CLIP, B-VQA, and 3-in-1 scores for measuring the generation of spatial/non-spatial relationships, attributes, and complex scenes, respectively. Following [35], we use the UniDet [39], CLIP [6], and BLIP [7] to evaluate these results. Our DisCo surpasses all compared T2I methods, confirming the efficacy of scene graphs in depicting complex scenes.

Table 2: **Relationship and attribute generation** compared with text-to-image methods on T2I-CompBench [35].

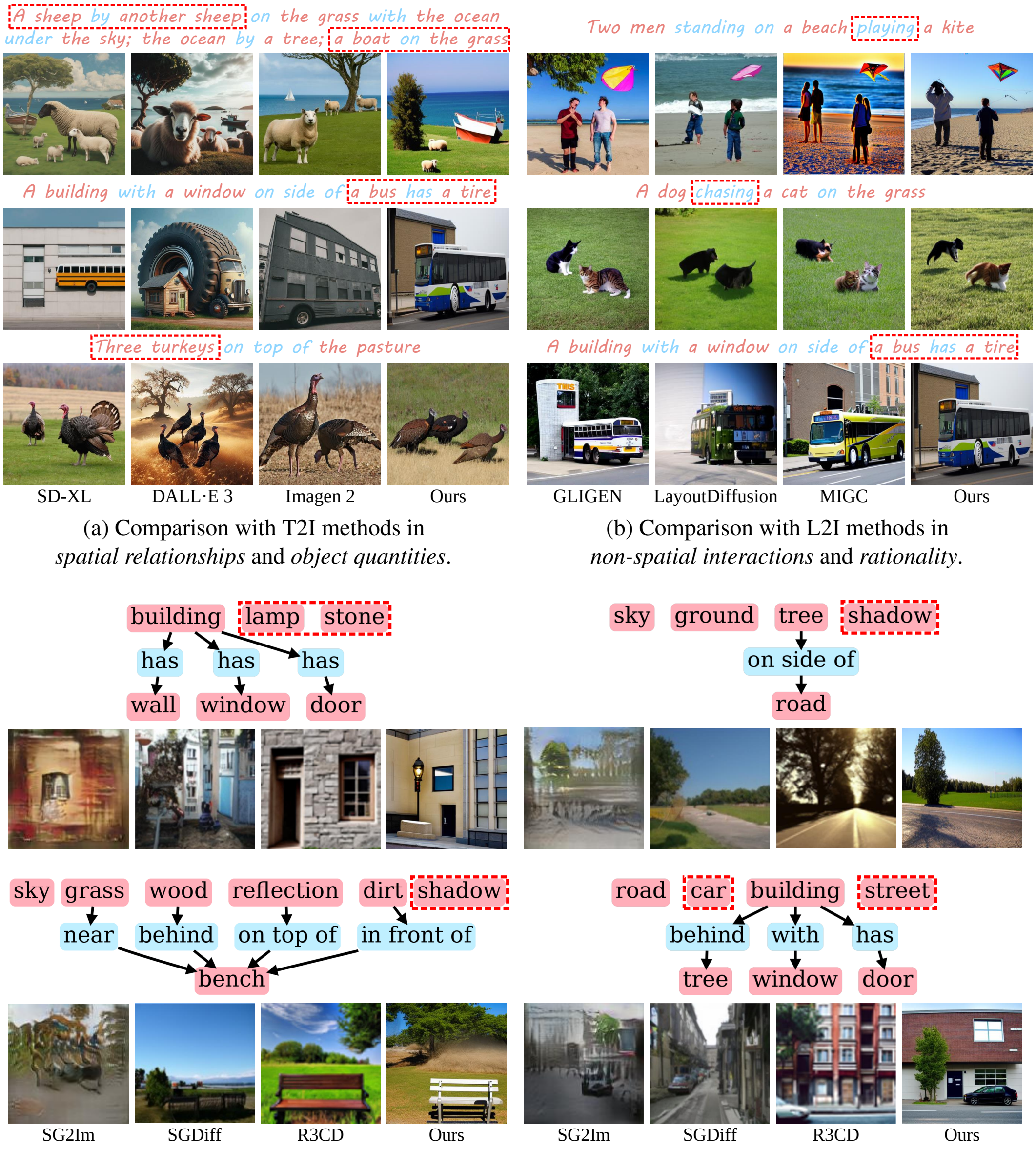
Method	**UniDet**	**CLIP**	**B-VQA**	**3-in-1**
SD-v1.4 [20]	0.1246	0.3079	0.3765	0.3080
SD-v2 [20]	0.1342	0.3127	0.5065	0.3386
Composable [36]	0.0800	0.2980	0.4063	0.2898
Structured [37]	0.1386	0.3111	0.4990	0.3355
Attn-Exct [38]	0.1455	0.3109	0.6400	0.3401
GORS [35]	0.1815	0.3193	0.6603	0.3328
DisCo (ours)	**0.2376**	**0.3217**	**0.6959**	**0.4143**

Table 3: **User study**. The score quantifies the user evaluation (i.e., relationships, quantities, and generation quality) of the alignment between the given prompt and the generated image.

Method	SD-XL [40]	DALL·E 3 [3]	Imagen 2 [41]	GLIGEN [12]	LD [10]	MIGC [13]	SG2Im [17]	SGDiff [16]	R3CD [14]	DisCo (ours)
Score	0.6684	0.5944	0.5637	0.6549	0.6200	0.7055	0.3783	0.4717	0.6928	**0.8533**

User Study. We conduct a user study by recruiting 50 participants from Amazon Mechanical Turk. We randomly select 8 prompts for each method, resulting in 80 generated images. We ask participants to score each generated image independently based on the image-prompt alignment. The worker can choose a score from $\{1, 2, 3, 4, 5\}$ and we normalize the scores by dividing them by 5. We then compute the average score across all images and all workers. The results are presented in the Table 3. Our method is favored by most participants in terms of generation rationality and controllability.

[1] https://huggingface.co/runwayml/stable-diffusion-v1-5


A sheep by another sheep on the grass with the ocean under the sky; the ocean by a tree; a boat on the grass
A building with a window on side of a bus has a tire
Three turkeys on top of the pasture
SD-XL
DALL·E 3
Imagen 2
Ours
Two men standing on a beach playing a kite
A dog chasing a cat on the grass
A building with a window on side of a bus has a tire
GLIGEN
LayoutDiffusion
MIGC
Ours
building
lamp
stone
has
has
has
wall
window
door
sky
ground
tree
shadow
on side of
road
sky
grass
wood
reflection
dirt
shadow
near
behind
on top of
in front of
bench
road
car
building
street
behind
with
has
tree
window
door
SG2Im
SGDiff
R3CD
Ours
SG2Im
SGDiff
R3CD
Ours

(a) Comparison with T2I methods in *spatial relationships* and *object quantities*.

(b) Comparison with L2I methods in *non-spatial interactions* and *rationality*.

(c) Comparison with SG2I methods in *independent node inference* and *generation quality*.

Figure 5: **Qualitative Comparisons** with **(a)** text-to-image (T2I) (SableDiffusion-XL [40], DALL·E 3 [3], and Imagen 2 [41]), **(b)** layout-to-image (L2I) (GLIGEN [12], LayoutDiffusion [10], and MIGC [13]), and **(c)** scene-graph-to-image (SG2I) (SG2Im [17], SGDiff [16], and R3CD [14]) methods.

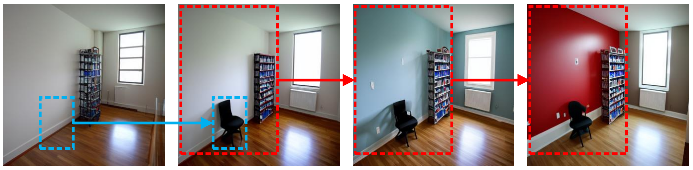
Table 4: **Ablation study** for overall architecture. SL-VAE w/o D_s means independent use of $\mathcal{O}$.

Method	G2I-ACC ↑	I2G-ACC ↑
Layout (D_l)	70.3	70.5
Semantics (D_s)	71.1	71.5
SL-VAE (w/o D_s)	72.9	72.8
SL-VAE (D_l + D_s)	**73.9**	**74.3**

Table 5: **Ablation study** for attention mechanism. Vanilla attention means off-the-shelf T2I attention.

Attention Type	IS ↑	FID ↓
Vanilla attention	17.2	29.1
CMA (w/o mask $\mathbb{M}$)	17.9	28.4
CMA (union MLP)	19.8	22.0
CMA (separate MLP)	**22.3**	**21.9**

Qualitative Comparisons. Figure 5 visualizes the results of the methods conditioned by text, layout, or scene graph, showcasing our advantages in generating rationality and controllability: (**i**) *Comparison with the text-to-image (T2I) methods*. In Figure 5 (a), we present the superiority of



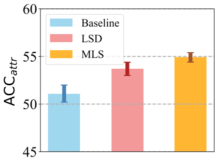
(a) unmodified (b) w/ NA (c) w/ AC (blue) (d) w/ AC (red)

Figure 6: **Illustration** of object-level Node Addition (NA) and Attribute Control (AC) in the scene. From left to right: **(a)** the image generated by the unmodified scene graph; **(b)** the chair addition; **(c)** the blue-colored wall; and **(d)** the red-colored wall.

the disentangled structured scene graph over linear text for representing complex scenes. Firstly, we resolve ambiguity in textual relationships and semantics by employing layout and semantic disentanglement within the scene graph. For example, our DisCo clarifies the relationship between "boat" and "grass" in the first line, as well as the semantics of "bus" and "building" in the following line. Additionally, the samples in the first and last lines also showcase our capacity to generate the specified quantities of objects precisely. (**ii**) *Comparison with the layout-to-image (L2I) methods*. We also demonstrate that our DisCo outperforms the diffusion methods relying on manually crafted scene layout representations, as illustrated in Figure 5 (b). While the L2I method struggles to model non-spatial interactions (such as "playing" in the first line and "chasing" in the second line), our DisCo addresses this challenge using disentangled object interactive semantics. Furthermore, by establishing semantics among objects derived from their relationships, we prevent independent generation instances that rely solely on the layout, exemplified by the "bus" and "tire" in the last line. (**iii**) *Comparison with the scene-graph-to-image (SG2I) methods*. The SG2I visualization results of different methods are showcased in Figure 5 (c). Our DisCo significantly improves the quality of SG2I generation, particularly for independent nodes. The proposed layout and semantics disentanglement technique effectively capture both the spatial and interactive information of independent nodes. Taking the "lamp" and "stone" in the first image and the "shadow" in the second image as illustrations, these entities are neglected by previous methods, whereas our DisCo not only retains their semantic relevance but also infers their appropriate spatial placement within the scene. We also demonstrate the generalizable generation under consistency for graph manipulation (i.e., node addition and attribute control) in SG2I tasks, as shown in Figure 6.

Table 6: **Ablation study** for Multi-Layer Sampler (MLS).

Method	**IS ↑**	**FID ↓**
Baseline (w/o MLS)	20.5	23.0
w/ LSD [25]	21.1	22.7
w/ MLS (Ours)	**22.3**	**21.9**



Ablation Study. Table 4 explores the overall architecture by evaluating the alignment of the generated image with the objects and relationships depicted in the input scene graph. Following SGDiff [16], we conduct this analysis by graph-to-image (G2I) and image-to-graph (I2G) retrieval experiments. Note that "w/o D_s" means processing each node embedding by MLP independently,instead of obtaining object interactive semantics through D_s. We observe that the spatial layout and interactive semantics collaborate to boost both retrieval tasks. These results demonstrate the effectiveness of integrating explicit spatial relations with implicit interactive semantics. In Table 5, we study the impact of different attention mechanisms. We inject graph conditions using different mechanisms: (a) Vanilla attention mechanism in the T2I diffusion model without our CMA; (b) CMA without attention mask $\mathbb{M}$; (c) CMA with a union MLP after concatenating object and attribute embeddings; and (d) CMA with two separate MLPs before concatenating object and attribute embeddings. We found that CMA, which fuses separate encoding of object and attribute embeddings, significantly enhances the overall generation performance. Table 6 presents the Multi-Layer Sampler (MLS) ablation results, confirming its enhancement over the baseline and LSD [29]. In contrast to LSD, which randomly scrambles layouts, the proposed MLS naturally leverages a variety of coherent layouts and semantics produced by SL-VAE. Moreover, the increase in ACC$_{attr}$ scores also indicates that MLS facilitates controllability, especially in attribute control, while ensuring generation quality.

5 Related Works

Diffusion Models. Diffusion models (DMs) [2, 20, 31, 42] have achieved great success in high-quality image generation. The essence of DMs lies in estimating image distributions by iterative denoising noise-corrupted image, showcasing the superiority over VAEs [43, 44] and GANs [45] in training stability and likelihood estimation. To further explore the controllability of DMs, considerable efforts have been devoted to conditional generation based on DMs. Benefiting from the naturalness of language [6] and the advancements of vision-language foundation models [6, 7, 8, 9], numerous text-to-image DMs [40, 41, 46] are beginning to emerge, facilitating explicit control of the corresponding semantics and style. However, the expressive capacity of linear text is limited. Therefore, many studies also endeavor to bolster global control through supplementary conditions, such as depth [46, 47], layout [10, 11, 12, 13], segmentation map [46, 48], and scene graph [14, 15].

Image Generation from Scene Graphs. Scene graphs are structured scene representations, where nodes represent objects and edges represent relationships between objects [17, 27]. Given the superiority of scene graphs over linear text in delineating multiple objects and their intricate relationships [27, 49, 50], many studies investigate image generation from scene graphs. These approaches typically fall into two categories: layout-based and semantics-based methods. Layout-based methods [17, 18, 19, 28, 36, 51] initially map scene graphs to coarse scene layouts comprising multiple bounding boxes and further refine these layouts to images with a layout-to-image model (e.g., LayoutDiffusion [10]). While the layout depicts spatial relationships, it fails to capture abstract relationships within the scene, leading to a lack of object interaction. Another branch is semantics-based methods [14, 16, 29, 52, 53], which focus on graph understanding by directly encoding semantic information from the scene graph. Nevertheless, these methods have limitations in addressing independent scene nodes, leading to issues like entity loss and unreasonable placement. In this paper, we propose a compositional image generation that leverages the layout and semantics derived from the scene graph representation. We complement explicit layout and implicit semantics to enhance the understanding of the diffusion model for scene graphs. Additionally, to improve the controllability in the scene-graph-to-image task, we also attain generalizable generation for object-level graph manipulation (i.e., node addition and attribute control).

6 Limitations

The proposed CMA injects object-level information into the diffusion model via masks from the layout, effectively mitigating semantic ambiguity and limiting attribute leakage. In scenarios involving object overlap, the proposed CMA inhibits direct interaction between the visual token and the object embedding along with its attributes. Nonetheless, the attribute information from the visual token inadvertently leaks into the overlapping region in subsequent layers. Hence, there may be attribute leakage among the objects, as shown in Figure 7.

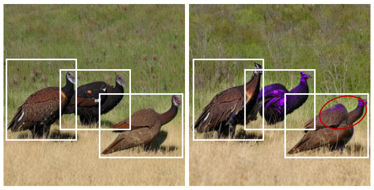


Figure 7: **Qualitative limitations** on attribute leakage of overlapping.

7 Conclusion

In this study, we leverage the disentangled textual scene graph representation to condition the diffusion process for generating complex scene images. The innovation of our framework lies in utilizing the VAE for scene relationship modeling and diffusion model (DM) for the composite visual generation. To comprehensively capture spatial relationships and non-spatial interactions within scenes, we introduce the Semantics-Layout Variational AutoEncoder (SL-VAE) for deriving diverse layouts and semantics from a single scene graph. Building upon them, we propose the Compositional Masked Attention (CMA) integrated with DM, which guides the de-noising trajectory by compositing extracted object-level graph information with fine-grained attributes. We also introduce a Multi-Layer Sampler (MLS) to preserve the main visual content while modifying the input scene graph. Extensive experiments demonstrate that our framework outperforms current methods conditioned by text, layout, or scene graph in relationship modeling and controllability.

Acknowledgments

This research is supported by the National Natural Science Foundation of China [Grant 62302246] and the Zhejiang Provincial Natural Science Foundation of China [Grant LQ23F010008]. We also express our sincere gratitude to the AI Computing Center at the Eastern Institute of Technology for their valuable support and assistance.

References

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:6840–6851, 2020. 1, 3

[2] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:8780–8794, 2021. 1, 9

[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*, 2:8, 2023. 1, 2, 7, 8

[4] 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. 1

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

[6] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 8748–8763, 2021. 1, 2, 3, 4, 7, 10

[7] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 12888–12900, 2022. 1, 7, 10

[8] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 19730–19742, 2023. 1, 10

[9] Kecheng Zheng, Yifei Zhang, Wei Wu, Fan Lu, Shuailei Ma, Xin Jin, Wei Chen, and Yujun Shen. Dreamlip: Language-image pre-training with long captions. *arXiv preprint arXiv:2403.17007*, 2024. 1, 10

[10] Guangcong Zheng, Xianpan Zhou, Xuewei Li, Zhongang Qi, Ying Shan, and Xi Li. Layoutdiffusion: Controllable diffusion model for layout-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22490–22499, 2023. 1, 2, 7, 8, 10

[11] Jiaxin Cheng, Xiao Liang, Xingjian Shi, Tong He, Tianjun Xiao, and Mu Li. Layoutdiffuse: Adapting foundational diffusion models for layout-to-image generation. *arXiv preprint arXiv:2302.08908*, 2023. 1, 10

[12] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22511–22521, 2023. 1, 7, 8, 10

[13] Dewei Zhou, You Li, Fan Ma, Zongxin Yang, and Yi Yang. Migc: Multi-instance generation controller for text-to-image synthesis. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 1, 7, 8, 10

[14] Jinxiu Liu and Qi Liu. R3cd: Scene graph to image generation with relation-aware compositional contrastive control diffusion. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 38, pages 3657–3665, 2024. 1, 2, 6, 7, 8, 10

[15] Azade Farshad, Yousef Yeganeh, Yu Chi, Chengzhi Shen, Böjrn Ommer, and Nassir Navab. Scenegenie: Scene graph guided diffusion models for image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 88–98, 2023. 1, 2, 6, 7, 10

[16] Ling Yang, Zhilin Huang, Yang Song, Shenda Hong, Guohao Li, Wentao Zhang, Bin Cui, Bernard Ghanem, and Ming-Hsuan Yang. Diffusion-based scene graph to image generation with masked contrastive pre-training. *arXiv preprint arXiv:2211.11138*, 2022. 1, 2, 6, 7, 8, 9, 10

[17] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1219–1228, 2018. 1, 2, 4, 6, 7, 8, 10

[18] Oron Ashual and Lior Wolf. Specifying object attributes and relations in interactive scene generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4561–4569, 2019. 2, 4, 6, 10

[19] Roei Herzig, Amir Bar, Huijuan Xu, Gal Chechik, Trevor Darrell, and Amir Globerson. Learning canonical representations for scene graph to image generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 210–227. Springer, 2020. 2, 6, 10

[20] 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, 2022. 3, 6, 7, 9, 14, 16

[21] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022. 3, 7

[22] Andrew Luo, Zhoutong Zhang, Jiajun Wu, and Joshua B Tenenbaum. End-to-end optimization of scene layout. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3754–3763, 2020. 4

[23] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:7537–7547, 2020. 5

[24] Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. $p+$: Extended textual conditioning in text-to-image generation. *arXiv preprint arXiv:2303.09522*, 2023. 5

[25] Jiawei Ren, Mengmeng Xu, Jui-Chieh Wu, Ziwei Liu, Tao Xiang, and Antoine Toisoul. Move anything with layered scene diffusion. *arXiv preprint arXiv:2404.07178*, 2024. 6, 9, 14, 15

[26] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1209–1218, 2018. 6, 7

[27] 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 (IJCV)*, 123:32–73, 2017. 6, 10

[28] Yikang Li, Tao Ma, Yeqi Bai, Nan Duan, Sining Wei, and Xiaogang Wang. Pastegan: A semi-parametric method to generate image from scene graph. *Advances in Neural Information Processing Systems (NeurIPS)*, 32, 2019. 6, 10

[29] Yang Wu, Pengxu Wei, and Liang Lin. Scene graph to image synthesis via knowledge consensus. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 37, pages 2856–2865, 2023. 6, 9, 10

[30] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[31] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 7, 9

[32] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in Neural Information Processing Systems (NeurIPS)*, 29, 2016. 7

[33] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017. 7

[34] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, 2016. 7

[35] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *Advances in Neural Information Processing Systems (NeurIPS)*, 36:78723–78747, 2023. 7

[36] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 423–439. Springer, 2022. 7, 10

[37] Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. *arXiv preprint arXiv:2212.05032*, 2022. 7

[38] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics (TOG)*, 42(4):1–10, 2023. 7

[39] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Simple multi-dataset detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7571–7580, 2022. 7

[40] 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. 7, 8, 10

[41] 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 (NeurIPS)*, 35:36479–36494, 2022. 7, 8, 10

[42] 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. 9

[43] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 9

[44] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. *Advances in Neural Information Processing Systems (NeurIPS)*, 28, 2015. 9

[45] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 9

[46] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3836–3847, 2023. 10

[47] Lezhong Wang, Jeppe Revall Frisvad, Mark Bo Jensen, and Siavash Arjomand Bigdeli. Stereodiffusion: Training-free stereo image generation using latent diffusion models. *arXiv preprint arXiv:2403.04965*, 2024. 10

[48] Ziqi Huang, Kelvin CK Chan, Yuming Jiang, and Ziwei Liu. Collaborative diffusion for multi-modal face generation and editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6080–6090, 2023. 10

[49] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3668–3678, 2015. 10

[50] Subarna Tripathi, Anahita Bhiwandiwalla, Alexei Bastidas, and Hanlin Tang. Using scene graph context to improve image generation. *arXiv preprint arXiv:1901.03762*, 2019. 10

[51] Yilun Du, Conor Durkan, Robin Strudel, Joshua B Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Sussman Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 8489–8510, 2023. 10

[52] Ruijun Li, Weihua Li, Yi Yang, Hanyu Wei, Jianhua Jiang, and Quan Bai. Swinv2-imagen: Hierarchical vision transformer diffusion models for text-to-image generation. *Neural Computing and Applications*, pages 1–16, 2023. 10

[53] Ruichen Wang, Zekang Chen, Chen Chen, Jian Ma, Haonan Lu, and Xiaodong Lin. Compositional text-to-image synthesis with attention map control of diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 38, pages 5544–5552, 2024. 10

A Appendix

This appendix is organized as follows: Section A.1 provides more details about Semantics-Layout Variation AutoEncoder; Section A.2 introduces the Stable Diffusion and its attention mechanism; Section A.3 describes the implementation of the Multi-Layer Sampler in detail; Section A.4 covers more ablation studies; Section A.5 presents more qualitative results, including comparison visualization and graph manipulation; Section A.6 delves into the broader societal impacts of this work. The core script is zipped and attached to the supplementary material.

A.1 Semantics-Layout Variation AutoEncoder

Recall that we apply the triplet-GCN-based CVAE architecture in Section 3.1. Each triplet-GCN layer in the encoder and decoder takes the node and edge embeddings. Specifically, the GCN_l mentioned in the paper uses two cascading MLPs $\{\mathrm{mlp}_1, \mathrm{mlp}_2\}$ to deal with node and edge embeddings:

$$(\psi_i^l, \phi_{ij}^{l+1}, \psi_j^l) = \mathrm{mlp}_1(\phi_i^l, \phi_{ij}^l, \phi_j^l), l \in \{0, \dots, L-1\}, \tag{12}$$

$$\phi_i^{l+1} = \psi_i^l + \mathrm{mlp}_2(\mathrm{avg}(\psi_j^l \mid j \in \mathcal{N}_{\mathcal{E}}(o_i))), \tag{13}$$

where l denotes the layer index of encoder or decoder, $\mathcal{N}_{\mathcal{E}}$ denotes the neighbor index set for each node, avg denotes the average pooling operation, ϕ and ψ denote intermediate features. Hence, mlp_1 conducts message passing among interconnected nodes and updates the edge features, while mlp_2 aggregates features from all neighboring nodes and updates its features. For graph union encoder, we let $(\phi_i^0, \phi_{ij}^0, \phi_j^0) = (\mathcal{O}_i, \mathcal{E}_{ij}, \mathcal{O}_j)$. The last embedding ϕ_i^L is parameterized to the Gaussian distribution $Z \sim \mathcal{N}(\mu, \sigma)$, where $\mu, \sigma \in \mathbb{R}^{D_z}$ output by two additional MLPs and D_z denotes the dimensional of latent space for node embedding.

A.2 Diffusion with Compositional Masked Attention

Stable Diffusion [20] is one of most popular text-to-image model. As described in Section 2.1, Stable Diffusion uses a U-Net $\boldsymbol{\epsilon}_\theta$ composed of convolution and transformer to estimate noise. The transformer includes two attention mechanisms, namely Cross-Attention, and Self-Attention.

Cross-Attention Layer. Text prompts are mapped to sequence embeddings by CLIP text encoder and integrated into UNet via Cross-Attention to guide the de-noising trajectory:

$$Attention(Q_{visual}, K_{text}, V_{text}) = softmax(\frac{Q_{visual} K_{text}^T}{\sqrt{d}}) \cdot V_{text} \tag{14}$$

where Q_{visual} denotes the Query from the visual token of the UNet, K_{text} and V_{text} denotes Key and Value from text embeddings, all of which are projected by linear layers, d denotes the dimension of Q_{visual}, K_{text}, and V_{text}.

Self-Attention Layer. Self-Attention captures self-related information within visual tokens:

$$Attention(Q_{visual}, K_{visual}, V_{visual}) = softmax(\frac{Q_{visual} K_{visual}^T}{\sqrt{d}}) \cdot V_{visual} \tag{15}$$

where Q_{visual}, K_{visual}, and V_{visual} separately represent the Query, Key, and Value in self-attention layers, which are projected by linear layers. The self-attention mechanism isolates the information flow between specific tokens by multiplying a mask $\mathbf{M}$ to the $Q_{visual} K_{visual}^T$. Since $\mathbf{M}$ is applied before $softmax$, the value of the isolated position is set to negative infinity $-inf$.

Compositional Masked Attention Layer. Based on the attention mask $\mathbf{M}$ that depends on layout $\mathcal{B}$, the Compositional Masked Attention can be expressed as:

$$Attention(Q_{CMA}, K_{CMA}, V_{CMA}) = softmax(\frac{Q_{CMA} K_{CMA}^T \odot \mathbf{M}}{\sqrt{d}}) \cdot V_{CMA} \tag{16}$$

where Q_{CMA}, K_{CMA}, and V_{CMA} individually represent the Query, Key, and Value derived from $\mathcal{V} \otimes \hat{\mathcal{C}}$, achieved through linear layer projections. We insert our proposed Compositional Masked Attention (CMA) between self-attention and cross-attention layers.

A.3 Multi-Layer Sampler

Layered Scene Representation. We decompose a controllable scene containing N_o objects into N_o layers. Different from SceneDiffusion [25], our approach involves each layer incorporating not only separate latent code z_i and spatial layout b_i, but also integrating the interactive semantics s_i produced by the SL-VAE. Here we convert the layout parameter b_i to two parts: (1) a fixed *object-centric* binary mask $m_i \in \{0,1\}^{c\times w\times h}$ to solely show the geometric property of the object, and (2) a two-element offset $p_i = \{\mu_i, \upsilon_i\}$ to solely indicate its spatial locations, with μ_i and υ_i defining the horizontal and vertical movement range. We sample Gaussian noise individually for the initial latent code of each layer, i.e., $\mathcal{Z} = \{z_i^{(T)} \sim \mathcal{N}(0,1)\}_{i=1}^{N_o}$. Then we utilize the layout-converted non-overlapping masks $\{l_i\}_{i=1}^{N_o}$ to derive the aggregated latent code $\boldsymbol{z}$ from various layers:

$$\boldsymbol{z}^{(t)} = \sum_{i=1}^{N_o} l_i \odot \overline{shift}(\boldsymbol{z}_i^{(t)}, p_i) \tag{17}$$

$$l_i = \overline{shift}(m_i, p_i) \prod_{j=1}^{N_i-1} (1 - \overline{shift}(m_j, p_j)), \tag{18}$$

where $\odot$ denotes element-wise multiplication, and $\overline{shift}(x,p)$ denotes spatially shifting the values of x in the direction of p.

Multi-Layer Generation. We introduce the Multi-Layer Sampler that matches our diverse layout and semantic simulation. In contrast to SceneDiffusion [25] which scrambles the reference layouts randomly, we sample additional N_l layouts and semantics by the proposed SL-VAE. On the one hand, the SL-VAE ensures that the generated scene layout is reasonable. On the other hand, we take full advantage of the paired object-level *(layouts, semantics)*. Specifically, the denoising scheme consists of four steps:

(a) Sampling additional N_l layouts $\{\mathcal{B}_n = \{b_{n,i}\}_{i=1}^{N_o}\}_{n=1}^{N_l}$ and semantics $\{\mathcal{S}_n = \{s_{n,i}\}_{i=1}^{N_o}\}_{n=1}^{N_l}$ by the proposed SL-VAE. Note that N_l fixed seeds exist for the same scene graph. According to the description of the layered representation, we convert the layout to get offset $\{\mathcal{P}_n = \{p_{n,i}\}_{i=1}^{N_o}\}_{n=1}^{N_l}$.

(b) Aggregating latent codes from various layers in each scene:

$$\boldsymbol{z}_n^{(t)} = \sum_{i=1}^{N_o} l_i \odot \overline{shift}(\boldsymbol{z}_i^{(t)}, p_{n,i}) \tag{19}$$

(c) Estimating the noise $\hat{\boldsymbol{\epsilon}}_n^{(t)}$ from each aggregated latent code $\boldsymbol{z}_n^{(t)}$ and gets denoised aggregated latent code $\hat{\boldsymbol{z}}_n^{(t-1)} \in \{\hat{\boldsymbol{z}}_1^{(t-1)}, \ldots, \hat{\boldsymbol{z}}_{N_l}^{(t-1)}\}$:

$$\hat{\boldsymbol{\epsilon}}_n^{(t)} = \sum_{i=1}^{N_o} m_{n,i} \odot \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_n^{(t)}, E_{\text{CLIP}}(o_i), b_{n,i}, s_{n,i}, a_i, t), \tag{20}$$

where $m_{n,i}$ is the non-overlapping mask converted by the layout $b_{n,i}$.

(d) Updating the latent code of each layer by computing the weighted average of the N_l aggregated latent code

$$\boldsymbol{z}_i^{(t-1)} = \frac{\sum_{n=1}^{N_l} \overline{shift}(l_i \odot \hat{z}_n^{(t-1)}, -p_{n,i})}{\sum_{n=1}^{N_l} \overline{shift}(l_i, -p_{n,i})} \tag{21}$$

where $\overline{shift}(x,-p)$ denotes spatially shifting the values of x in the reverse direction of p.

A.4 More Ablation Studies

Graph Construction. We conduct ablation for graph construction in Table 7. We investigate the impact of different graph components (i.e., CLIP, Box, and Learnable Embeddings) by turning off each independently. We observe that each component improves the performance, all of which are crucial components presented in our DisCo.

Table 7: **Ablation study** for graph construction.

Graph Type	**IS ↑**	**FID ↓**
No CLIP Emb.	20.6	23.9
No Box Emb.	21.7	22.5
No Learnable Emb.	21.9	22.2

Computing Consumption. We demonstrate the impact of our proposed CMA on the computational complexity of the U-Net within the Stable Diffusion, as presented in Table 8. We use Floating Point

Table 8: **Ablation study** for computing consumption.

Method	FLOPs (G)	Params (M)	Time (ms)
SD-v1.5 [20]	677.5	859.4	37.9
DisCo	724.1	875.8	108.3

Operations (FLOPs), the number of parameters (Params), and inference time (Time) to measure computing consumption. The FLOPS and Time metrics are conducted by processing the tensor with a resolution of $2 \times 4 \times 64 \times 64$ on an NVIDIA A100 GPU. Our proposed DisCo significantly improves the controllability of the Stable Diffusion with a tolerable increase in computational cost.

A.5 More Visualization Results

Figure 8 showcases more generalizable generation results under consistency for graph manipulation (i.e., node addition and attribute control) in SG2I task. In Figure 9, 10, and 11, we present more visualization comparisons with the methods conditioned by text, layout, or scene graph, which demonstrates the superiority of our DisCo in terms of generation rationality and controllability.

A.6 Broader Impacts

We demonstrate the superiority of our DisCo over existing generation methods based on text, layout, and scene graphs, suggesting a potential beneficial influence on the realms of art creation and data synthesis. Nevertheless, there remains a concern regarding the possibility of generating malicious images or infringing copyright.

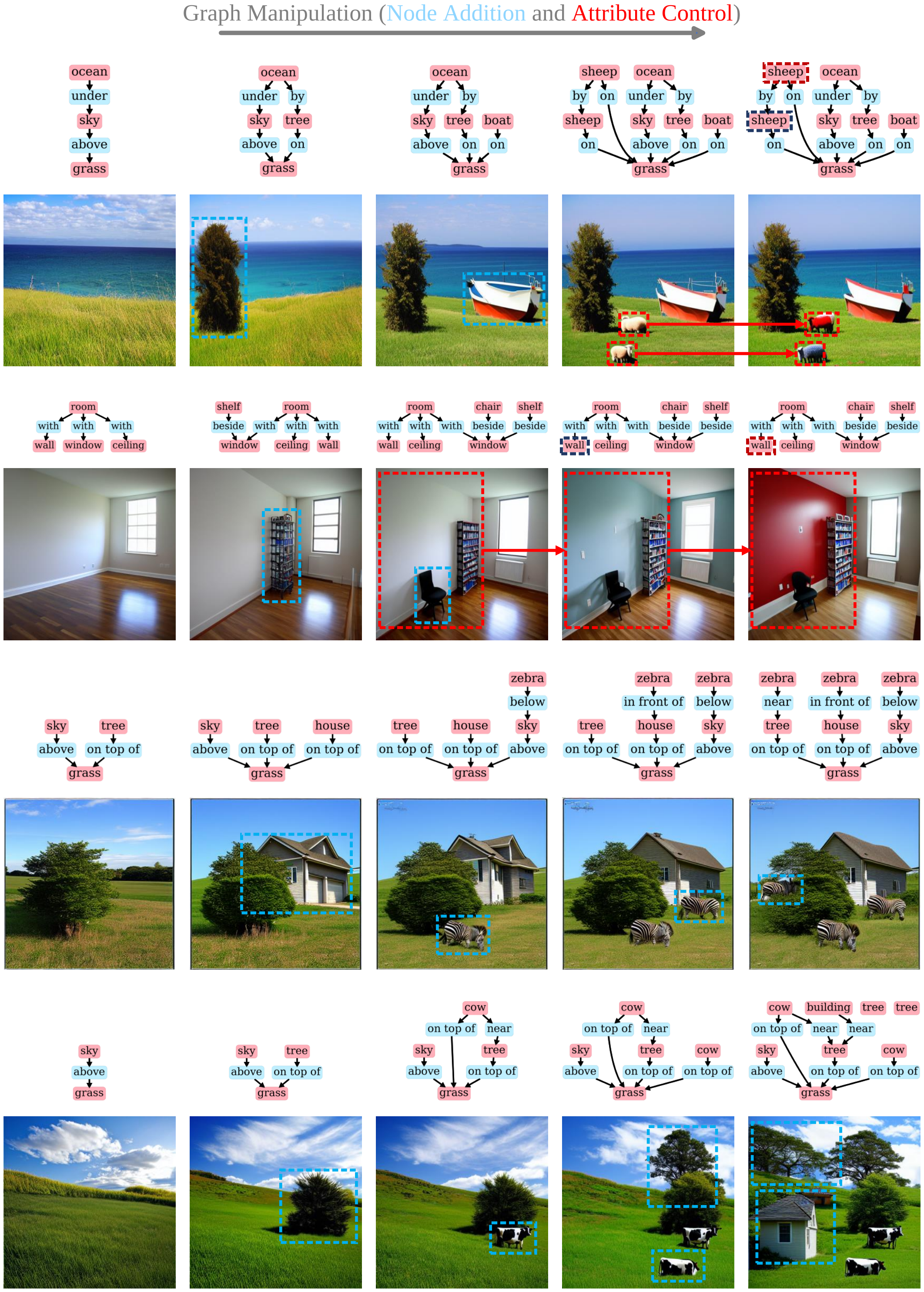

Graph Manipulation (Node Addition and Attribute Control)
ocean under sky above grass
ocean under by sky tree above on grass
ocean under by sky tree boat above on on grass
sheep ocean by on under by sheep sky tree boat on above on on grass
sheep ocean by on under by sheep sky tree boat on above on on grass
room with with with wall window ceiling
shelf room beside with with with window ceiling wall
room chair shelf with with with beside beside wall ceiling window
room chair shelf with with with beside beside wall ceiling window
room chair shelf with with with beside beside wall ceiling window
sky tree above on top of grass
sky tree house above on top of on top of grass
zebra below tree house sky on top of on top of above grass
zebra zebra in front of below tree house sky on top of on top of above grass
zebra zebra zebra near in front of below tree house sky on top of on top of above grass
sky above grass
sky tree above on top of grass
cow on top of near sky tree above on top of grass
cow on top of near sky tree cow above on top of on top of grass
cow building tree tree on top of near near sky tree cow above on top of on top of grass

Figure 8: **Generalizable Generation Samples** under Consistency for Graph Manipulation.

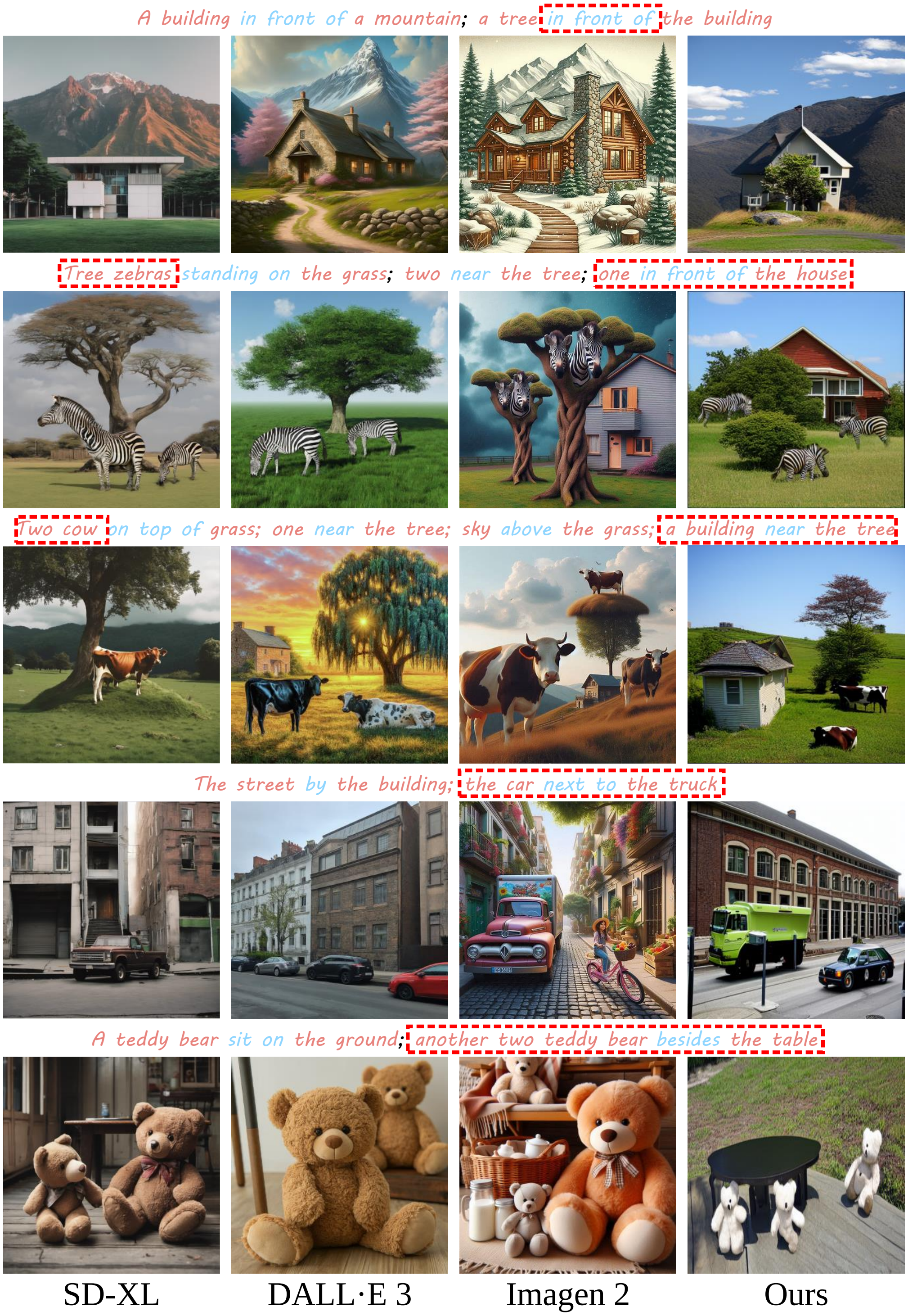

A building in front of a mountain; a tree in front of the building
Tree zebras standing on the grass; two near the tree; one in front of the house
Two cow on top of grass; one near the tree; sky above the grass; a building near the tree
The street by the building; the car next to the truck
A teddy bear sit on the ground; another two teddy bear besides the table
SD-XL
DALL·E 3
Imagen 2
Ours

Figure 9: **Qualitative Comparison** with Text-to-Image methods.

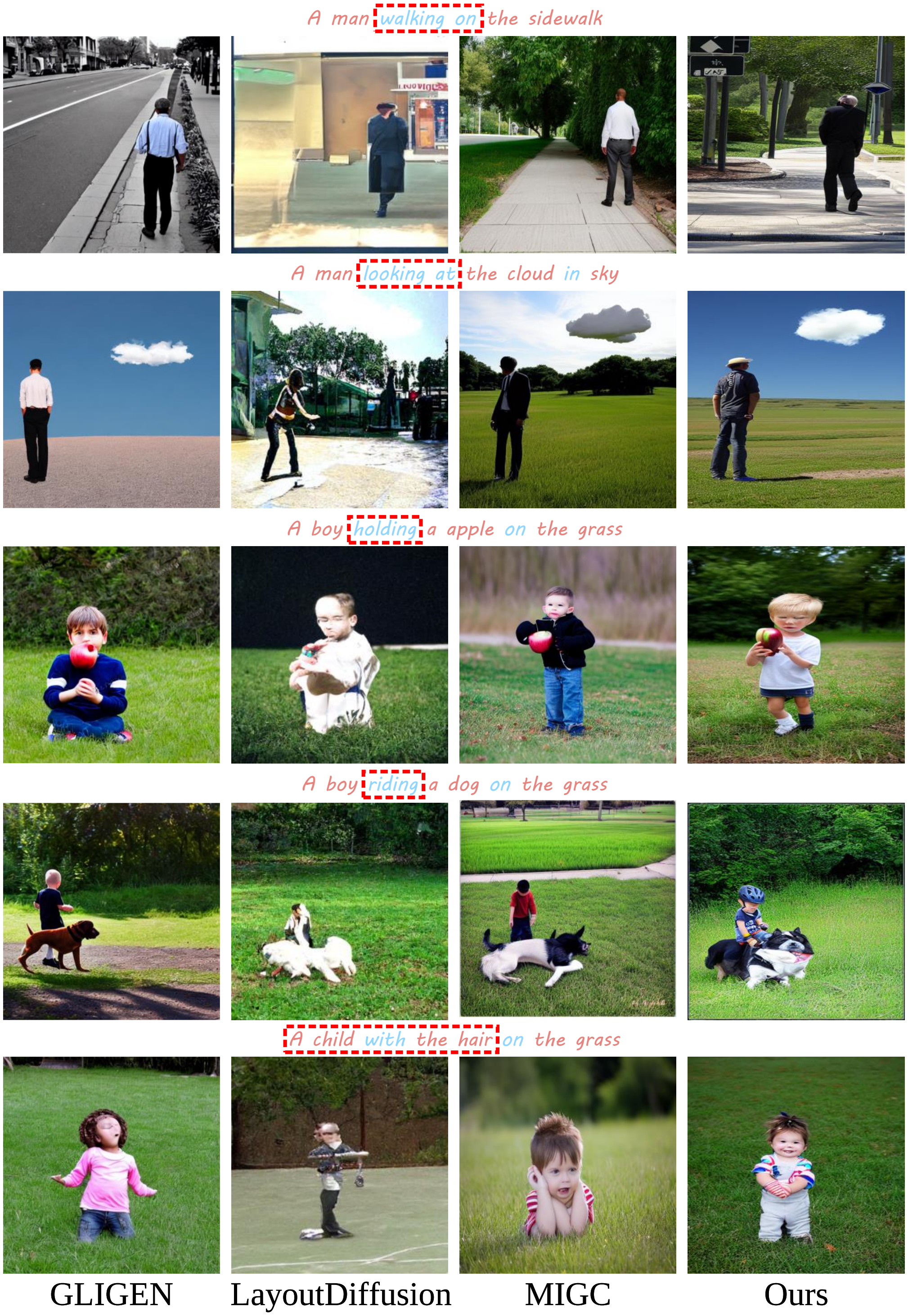

A man walking on the sidewalk
A man looking at the cloud in sky
A boy holding a apple on the grass
A boy riding a dog on the grass
A child with the hair on the grass
GLIGEN
LayoutDiffusion
MIGC
Ours

Figure 10: **Qualitative Comparison** with Layout-to-Image methods.

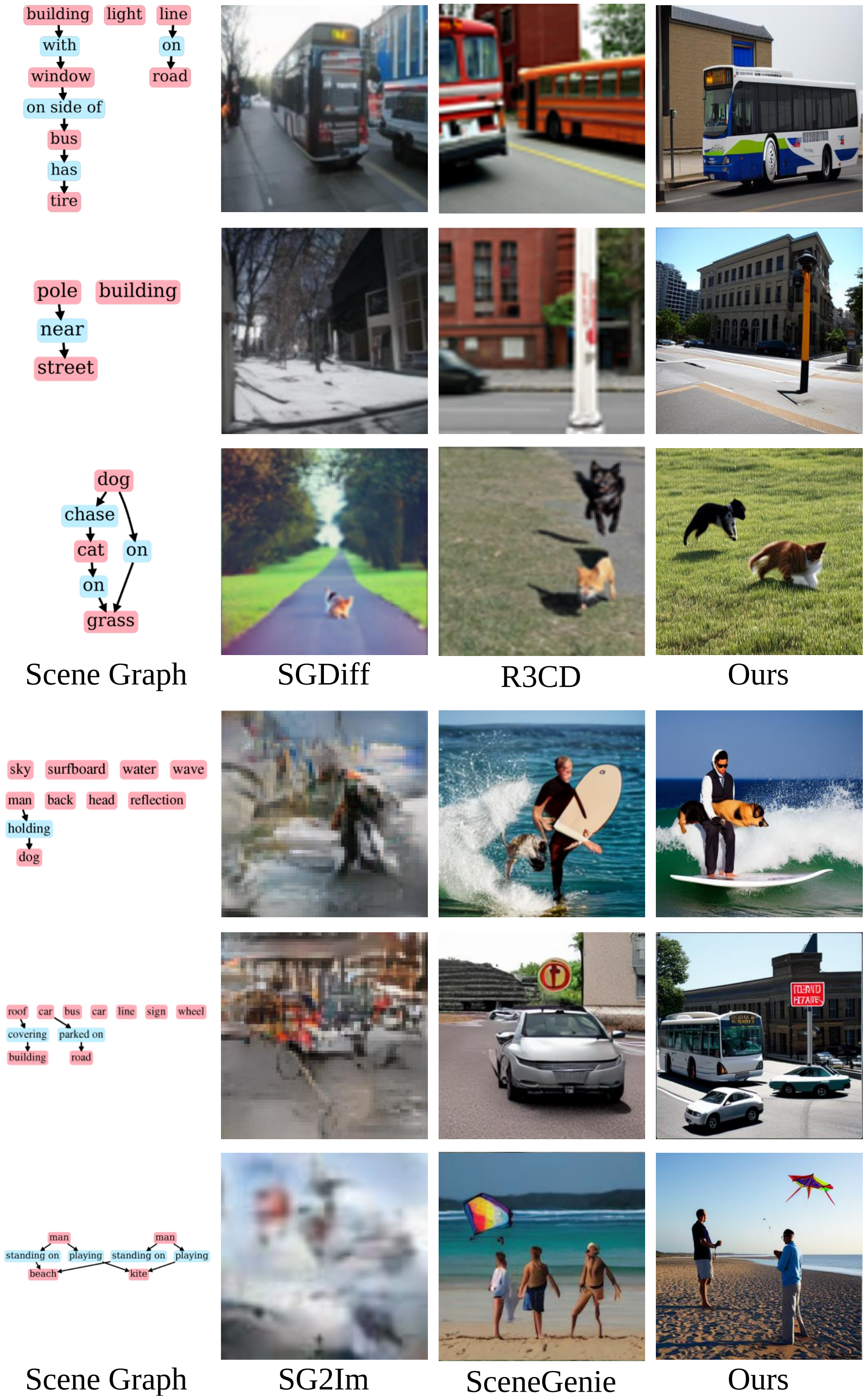


Figure 11: **Qualitative Comparison** with Scene-Graph-to-Image methods.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: We claim in the abstract and introduction that this paper studies generating complex images with structured scene graphs.

   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: The limitations are addressed in the Appendix, which will be included in the camera-ready version.

   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: [Yes]

Justification: All the theorems and formulas are clearly stated in Section 2 and 3 of the main paper, and complemented in Section A.1, A.2 and A.3 of the Supplemental Material. Besides, all theorems and assumptions related to this work are properly referenced.

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: We provide the implementation details for reproducibility in Section 4, and the project will be open-sourced.

   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 attach the core script and data preparation to the Supplemental Material. The complete code will be released in the camera-ready version, accompanied by detailed instructions for reproducibility.

   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: We provide the training and test details in Section 4.

   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: We report error bars suitably and correctly to achieve statistical significance in the paper.

   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: We provide sufficient information on the computer resources in Section 4.

   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: Our work conforms with the NeurIPS Code of Ethics in every respect.

   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 both potential positive and negative societal impacts in Section A.6 of the Appendix.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

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

11. **Safeguards**

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

    Answer: [Yes]

    Justification: Our model is heavily based on existing StableDiffusion, so our safeguards are the same as theirs.

    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: In Section 4, we properly credit all the public baselines and datasets utilized in this paper.

    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: The assets introduced in the paper are well documented.

    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: This paper does not involve crowdsourcing nor research with human subjects.

    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: This paper does not involve crowdsourcing nor research with human subjects.

    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.