# GRAPH2SCENE: RELATION-AWARE TEXT-TO-IMAGE GENERATION WITH GRAPH-GUIDED DIFFUSION MODEL

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Text-to-image diffusion models still struggle with complex prompts containing many elements and interactions. Existing attention-control methods mainly focus on controlling where elements appear, but often ignore how objects interact with each other. As a result, relation tokens describing interactions between elements may assign high attention to irrelevant regions, leading to missing or incorrect interactions. In this paper, we propose Graph2Scene, a training-free framework that treats scene graphs as a core control signal for diffusion generation. It represents elements as nodes and their relations as edges, and translates these element-relation dependencies into selective attention guidance. It grounds each element to its corresponding region and guides relational interactions among the involved elements. To achieve appropriate relation control without over-constraining the generation process, Graph2Scene further adapts the control duration to scene complexity and progressively relaxes graph guidance during denoising. Experiments on T2I-CompBench and ConceptMix show clear improvements in compositional generation, especially for object relations. On ConceptMix, Graph2Scene improves relationship accuracy from 0.59 to 0.84 over the strongest FLUX.1-based regional-control baseline. Results on LongBench-T2I further show its advantage on long and complex prompts. Code is available at `https://anonymous.4open.science/r/GraphExecutor-CF2F`.

## 1 INTRODUCTION

Text-to-image diffusion models (Rombach et al., 2022; BlackForestLabs, 2025; Chen et al., 2024b) have made remarkable progress in generating realistic images from natural-language prompts. As model capabilities continue to advance, user demands are increasingly shifting toward complex prompts, where a target image may contain multiple objects with diverse attributes, spatial arrangements, and interactions (Wu et al., 2024; Hoe et al., 2024a; Zhou et al., 2025). However, faithfully generating such complex scenes remains challenging, as the model must accurately capture the semantic dependencies among objects, attributes, and relations from long text prompts and translate them into the corresponding visual content.

In diffusion models, attention associates textual concepts with visual content during generation. Ideally, each text token should mainly influence the image region corresponding to its concept. However, in complex prompts, attention may spread to irrelevant regions, causing different concepts to interfere with each other. We refer to this undesired behavior as attention leakage (Wang et al., 2024; Dahary et al., 2024; Zhang et al., 2024). Attention leakage can lead to incorrect attributes, object interference, or missing elements during generation (Chefer et al., 2023; Wang et al., 2024; Rassin et al., 2023; Marioriyad et al., 2024). Existing attention-control methods alleviate this issue by introducing spatial constraints, such as bounding boxes or region masks, to restrict object-related tokens to their corresponding regions (Li et al., 2023; Wang et al., 2025; Xie et al., 2023; Liu et al., 2022; Yang et al., 2024).

Although existing methods can effectively place individual elements in their correct regions, they largely overlook attention leakage at the relation level. We conduct an experiment to verify that object-level layout control overlooks relation-level attention. Figure 1 compares (A) generation without attention control, (B) object-level layout control following (Chen et al., 2024a) (C) the



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

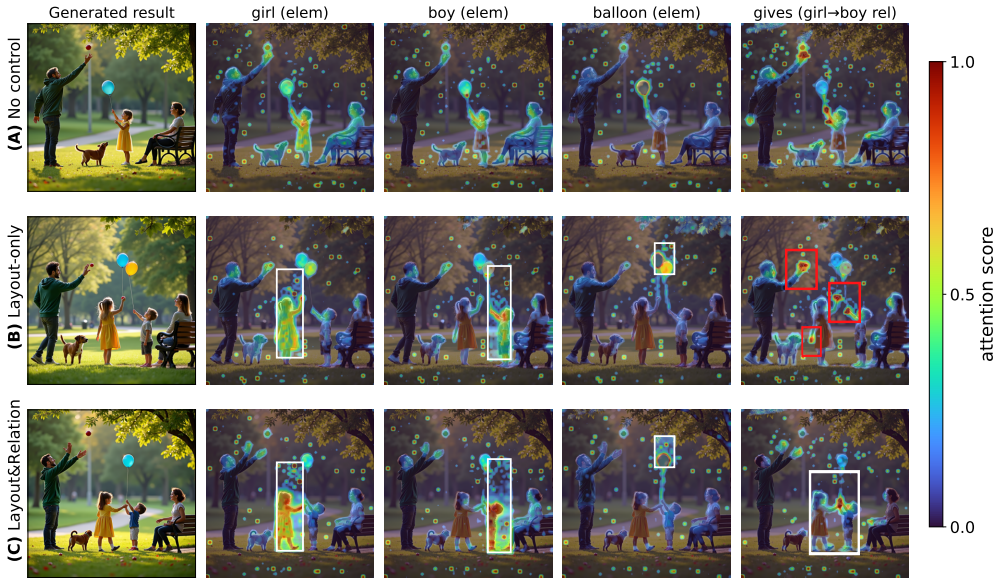



Figure 1: Cross-attention visualization on FLUX.1 for a selected interaction (*a girl gives a balloon to a boy*) extracted from a complex prompt. Heatmaps show the aggregated attention of object and relation tokens. Rows compare (A) no control, (B) layout-only control, and (C) layout and relation control. White boxes indicate the regions to which the corresponding text tokens are constrained by the proposed control, while red boxes highlight the attention leakage phenomenon.

same layout control, while restricting relation tokens to focus only on the related elements and their interaction regions. Without attention control (Figure 1(A)), the model fails to distinguish the regions of the girl and the boy, and the boy is missing. With layout control (Figure 1(B)), the girl, boy, and balloon are generated in the correct regions with proper attributes. However, the relation token *gives* is not explicitly controlled, and its attention mainly focuses on the hand of an unrelated man rather than the girl, boy, and balloon involved in the interaction (as highlighted by the red box in Figure 1(B)). As a result, the intended giving relation is not correctly generated. By restricting *gives* to attend to the related elements and their interaction region (Figure 1(C)), the intended relation becomes much clearer. This experiment demonstrates that object-level layout control alone is insufficient, and relation tokens also require explicit attention control.

Effectively controlling relations in complex scenes requires identifying which elements are involved in each interaction. Scene graphs naturally provide such a structure by representing elements as nodes and interactions as edges, making the dependencies in complex prompts explicit. However, using scene graphs to guide complex image generation raises two key challenges. First, a scene graph only describes these dependencies and cannot directly guide a diffusion model. We need to determine how to translate its nodes and edges into effective attention operations. Second, the attention control mechanism should adapt to different denoising stages (Chen et al., 2024c; Ma et al., 2024; Lee et al., 2025), while avoiding excessive intervention that may disturb the pretrained generation process.

To address these challenges, we propose **Graph2Scene**, a training-free framework that addresses relation control in complex text-to-image generation through two key designs. First, Graph2Scene introduces a graph-guided attention execution mechanism that converts scene graphs into structured attention operations. It explicitly models the correspondence between semantic concepts and visual regions: cross-attention associates relation concepts with their involved elements, while self-attention enables information propagation among graph-connected regions to reduce attention leakage and preserve object interactions. Second, Graph2Scene introduces a complexity-aware stage-wise execution strategy that estimates generation difficulty from attention patterns and adaptively determines when and how long graph control should be applied. This allows the framework to provide necessary guidance for challenging scenes while preserving the native generation ability of the pretrained model.

In summary, our contributions are as follows:

- We identify that faithful generation of complex scenes requires explicit coordination of attention among interacting objects, beyond the object grounding and spatial control emphasized by existing methods.
- Based on this observation, we propose **Graph2Scene**, a training-free framework that translates scene graphs into relation-aware attention control. It compiles element-relation dependencies into selective cross- and self-attention guidance and adaptively executes such guidance along the denoising trajectory according to scene complexity.
- Extensive experiments on T2I-CompBench, ConceptMix, and LongBench-T2I show that Graph2Scene consistently improves compositional generation, especially on complex relation-rich scenes, improving relationship accuracy on ConceptMix from 0.42 to 0.84 compared with the FLUX.1 model.

## 2 RELATED WORK

### 2.1 COMPOSITIONAL DIFFUSION GENERATION.

Text-to-image diffusion models(Rombach et al., 2022; BlackForestLabs, 2025; Chen et al., 2024b) can generate high-quality images from text. However, they often fail on compositional prompts with multiple objects, attributes, and relations. Common errors include missing objects, wrong attribute binding, and incorrect relations(Dahary et al., 2024; Marioriyad et al., 2024). This has motivated many studies on compositional diffusion generation. One line of work injects external layout signals into the attention. GLIGEN (Li et al., 2023) adds gated attention layers that take grounding boxes as input, and MS-Diffusion (Wang et al., 2025) trains an adapter that binds reference subjects to given regions. These methods pre-train small modules, and then steer where each object appears.

A second line of work directly constrains or optimizes the existing attention maps at inference time, without adding modules.Attend-and-Excitep (Chefer et al., 2023) strengthens the cross-attention responses of subject tokens during inference, preventing important concepts from being neglected in the generated image. Composable Diffusion (Liu et al., 2022) separately predicts the denoising direction for each concept and combines them at every diffusion step to generate all concepts together. Regional Prompting (Chen et al., 2025) splits the attention by region so each prompt attends to its own area. LLM-Blueprint (Gani et al., 2024) and RPG (Yang et al., 2024) further use an LLM to decompose complex prompts and plan object layouts before applying region-level generation or attention control. These methods reshape attention per object, but they treat each object on its own. Although these methods(Hoe et al., 2024b; Mei et al., 2026) consider interactions, they mainly focus on generating specific interaction patterns in simple scenes. They rely on predefined HOI categories or learned interaction representations, and cannot generalize to diverse relations among multiple elements in complex scenes.

Overall, existing methods mainly improve compositional generation through object grounding, attribute binding, or layout control. Their guidance is largely defined at the object level and does not explicitly model the interactions among multiple elements. As scene complexity increases, this object-wise control is insufficient to capture relation-dependent generation, leading to incorrect or missing interactions even when individual objects are properly placed.

### 2.2 SCENE GRAPHS FOR DIFFUSION MODELS

Scene graphs are also used to represent objects and relations in complex scenes. Existing methods mainly differ in how the graph is converted into a generation condition. SG2Im (Johnson et al., 2018) uses a GCN to encode the graph and predict object layouts. SGDiff (Yang et al., 2022) learns graph or relation embeddings. CanonicalSg2Im (Herzig et al., 2020) learns canonical graph representations to handle semantically equivalent graphs. SG-Adapter (Shen et al., 2025) uses graph triplets to refine text embeddings. R3CD (Liu & Liu, 2024) further models relations with a graph transformer and relation-aware attention objectives.

Most existing scene graph-based methods focus on scene graph-to-image generation, where scene graphs are directly used as inputs to graph-conditioned generators or trained graph encoders. In contrast, Graph2Scene does not train a new graph-conditioned generation model, but explores how to execute scene graph constraints on a pretrained text-to-image diffusion model through attention

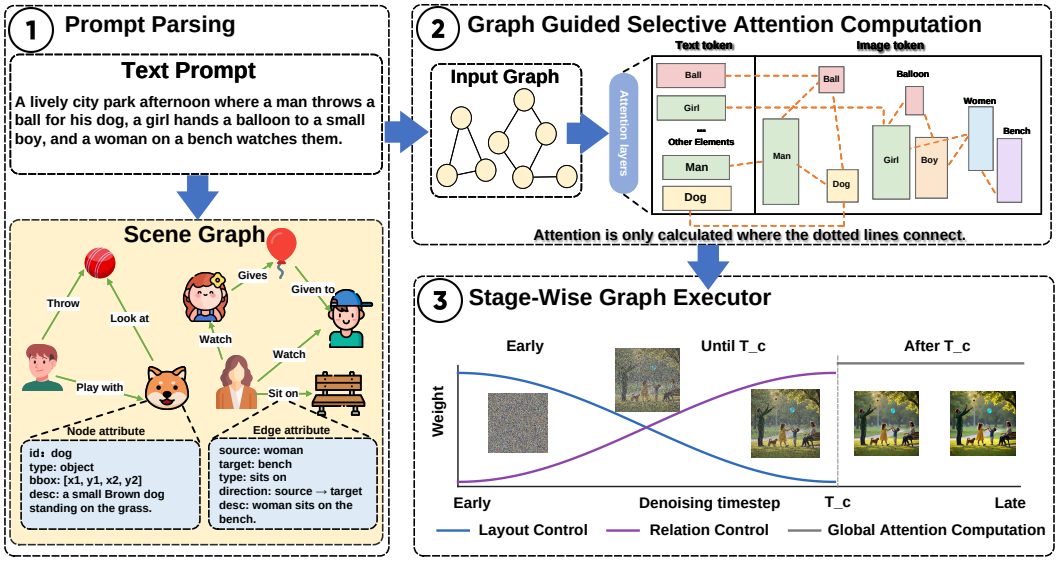



Figure 2: Workflow of Graph2Scene.

control. Therefore, in the main experiments, we mainly compare with existing compositional generation methods built upon pretrained T2I models, while providing additional comparisons with scene graph-based generation methods in Appendix A.3.

## 3 GRAPH2SCENE

As shown in Figure. 2, Graph2Scene consists of three main components. First, we construct a scene graph from a complex text prompt. Each element node contains its semantic description and predicted layout, while each relation edge connects the corresponding elements involved in an interaction. Second, we compile the scene graph into attention masks for the frozen Diffusion Transformer. In cross-attention, the masks control the association between text tokens and image regions, so that element descriptions attend to their target regions and relation tokens are bound to the regions of the elements involved in each interaction. In self-attention, the masks restrict information exchange according to the graph structure, allowing communication within each element region and between graph-connected regions, which further reduces attention leakage across unrelated regions. Third, we apply these graph-based controls along the denoising trajectory with a stage-wise schedule. The control duration is determined by the estimated scene complexity, allowing the framework to adjust the execution process across different denoising stages.

### 3.1 HIERARCHICAL SCENE GRAPH CONSTRUCTION

Graph2Scene represents a complex prompt as a scene graph of elements and their relations, making the structural dependencies in the prompt explicit. Specifically, we use a vision–language model (VLM) () to parse the prompt into a hierarchical representation with a global scene and individual element nodes. Each element node stores its semantic description and predicted layout information, including its bounding box. Each edge connects two element nodes and specifies their relation, such as holding or next to. The graph explicitly represents where each element should appear and how different elements should interact, providing structured guidance for attention computation during the subsequent denoising process. The detailed construction templates and examples are provided in Appendix A.1.

### 3.2 GRAPH GUIDED SELECTIVE ATTENTION COMPUTATION

To ensure that relational concepts interact with their corresponding elements, Graph2Scene introduces a customized attention mask mechanism for the MM-DiT backbone. The mask is constructed from the scene graph structure, including the topology of nodes and edges, as well as the layout and semantic information associated with each node and edge.

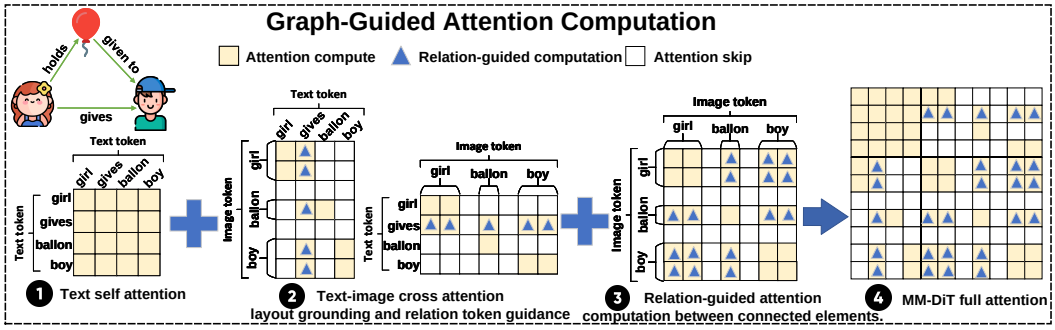



Figure 3: Illustration of graph-guided attention computation. Cross-attention associates element and relation tokens with the corresponding image regions, while self-attention allows information exchange within each region and between graph-connected regions.

As illustrated in Figure 3, Graph2Scene converts the scene graph into structured attention computation through a customized attention mask, which regulates information flow in the MM-DiT backbone. Graph2Scene performs graph-guided attention through two operations: text-image cross-attention and relation-guided self-attention. The **text-image cross-attention computation** (❷) associates text tokens with image tokens. Element tokens are connected to their corresponding image regions, while relation tokens are connected to the regions of all participating elements. This provides layout grounding and binds each relation to its related elements. The **relation-guided self-attention computation** (❸) controls information exchange among image tokens. Tokens within the same element region can interact with each other, while tokens from different regions can exchange information only when their corresponding elements are connected in the scene graph. Together with text self-attention (❶), these operations define the graph-guided attention computation, which is integrated into the full MM-DiT attention (❹).

Graph2Scene translates the scene graph into a customized attention mask $M_{\mathcal{G}}$, which regulates information flow during attention computation. The mask consists of two parts: a cross-attention mask $M_{\text{cross}}$, which controls the interaction between text tokens and image regions, and a self-attention mask $M_{\text{self}}$, which regulates information exchange among image tokens. Given a scene graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, each node $v \in \mathcal{V}$ represents an element with its associated text token set $\mathcal{T}_v$ and spatial region $\mathcal{R}_v$ predicted by the VLM. Each edge $e \in \mathcal{E}$ represents a relation with token set $\mathcal{T}_e$, where $\mathcal{V}_e$ denotes the set of element nodes involved in this relation. We denote the image tokens corresponding to the region of node $v$ as $\mathcal{X}_v$. Based on these graph associations, the attention mask determines the allowed token interactions during attention computation.

**Cross-attention mask.** The cross-attention mask controls the interaction between text tokens and image regions in both directions (❷). For each element node, its associated text tokens are connected to the corresponding image region. For each relation edge, the relation tokens are connected to the image regions of all participating nodes, allowing relational concepts to interact with the elements involved in the relation. Accordingly, we apply the same graph-guided connectivity to both text-to-image and image-to-text attention:

$$M_{\text{cross}}^{\text{T}\to\text{I}}(i,j) = M_{\text{cross}}^{\text{I}\to\text{T}}(j,i) = \begin{cases} 0, & \text{if } \big(t_i \in \mathcal{T}_v \wedge x_j \in \mathcal{X}_v\big) \vee \big(t_i \in \mathcal{T}_e \wedge x_j \in \bigcup_{v\in\mathcal{V}_e} \mathcal{X}_v\big), \\ -\infty, & \text{otherwise.} \end{cases} \tag{1}$$

**Self-attention mask.** The self-attention mask controls information exchange among image tokens (❸). In self-attention, each element region is first allowed to interact with itself to preserve its local visual consistency. Beyond self-region attention, we only enable interactions between regions whose corresponding elements are connected by a relation edge in the scene graph. This design maintains independent object representations while facilitating visual information exchange among relationally connected elements. Formally, the self-attention mask is defined as:

$$M_{\text{self}}(i,j) = \begin{cases} 0, & \text{if } \big(\exists v \in \mathcal{V} : x_i, x_j \in \mathcal{X}_v\big) \vee \big(\exists e \in \mathcal{E},\ u,v \in \mathcal{V}_e : x_i \in \mathcal{X}_u,\ x_j \in \mathcal{X}_v\big), \\ -\infty, & \text{otherwise.} \end{cases} \tag{2}$$

The overall attention computation with the graph-guided mask is formulated as:

$$\text{Attention}(Q, K, V; M_{\mathcal{G}}) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d}} + M_{\mathcal{G}}\right)V. \tag{3}$$

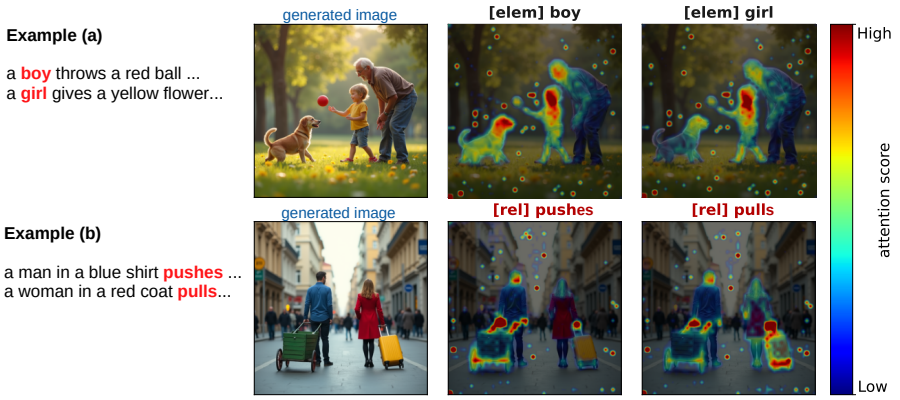



Figure 4: Visualization of two typical attention failures. Example (a) shows attention overlap between the *boy* and *girl* tokens, while example (b) shows attention leakage of the relation tokens *pushes* and *pulls*.

### 3.3 Stage-wise Graph Execution

Graph-guided attention control should be adaptive rather than applied with a fixed strategy for all prompts. First, continuous attention control throughout generation may harm the model's native generation ability (), so the control should be stopped in time once sufficient guidance has been provided. Second, relation control should follow layout control, since diffusion models typically establish object layouts before refining semantic relations and visual details (Lee et al., 2025; Pan et al., 2025).

We monitor the attention distributions of specific text tokens during generation to guide the adaptive adjustment of control strength. These attention distributions reveal two typical generation problems: attention overlap and attention leakage. Attention overlap occurs when the high-score attention of different tokens concentrates on largely the same spatial regions. Attention leakage occurs when the high-score attention distribution of a token extends beyond the region of its corresponding concept. Figure 4 illustrates both cases: in example (a), the attention maps of *boy* and *girl* overlap heavily, causing the girl to be missing, while in example (b), the relation tokens *pushes* and *pulls* fail to focus on their corresponding regions, leading to incorrect relation generation.

**Control Strength Adjustment.** Based on these attention behaviors, we first perform a warm-up generation without additional guidance and collect the attention distributions of the selected text tokens. We then measure the severity of attention overlap and attention leakage from these attention maps. For prompts with more severe attention overlap or leakage, Graph2Scene applies stronger and longer graph guidance. The detailed calculation is provided in Appendix A.5.

**Control Selection Strategy.** Graph2Scene progressively shifts the control focus from layout guidance to relation guidance during denoising. This design follows how diffusion models gradually form the layout first and refine semantics and interactions later. Since this transition is gradual and continuous (Zhang et al., 2025), we linearly decrease the layout guidance weight from 1 to 0 and increase the relation guidance weight from 0 to 1 over the determined control duration.

## 4 Experiment

### 4.1 Experiment Setting

**Datasets.** We evaluate Graph2Scene on two complementary benchmarks covering different levels of scene complexity. **T2I-CompBench** (Huang et al., 2023) evaluates compositional generation,

including attribute binding and object relationships, using relatively concise prompts. To assess performance on longer and more complex prompts, we follow the **ConceptMix** (Wu et al., 2024) protocol, which systematically controls scene complexity through the number of concepts $K$. We construct prompts with $K \in [8, 20]$ and introduce an additional *interaction* dimension to increase the density of object relations. This setting allows us to examine how generation performance changes as prompt complexity increases. We additionally report qualitative results on the long-prompt benchmark **LongBench-T2I** (Zhou et al., 2025) in Appendix A.6.

**Baselines.**We compare Graph2Scene with representative compositional and controllable generation methods, including **Composable Diffusion** (Liu et al., 2022), **GLIGEN** (Li et al., 2023), **Attend-and-Excitep** (Chefer et al., 2023), **MS-Diffusion** (Wang et al., 2025), **LLM-Blueprint** (Gani et al., 2024), **RPG** (Yang et al., 2024). Since these methods are built on different diffusion backbones, we follow their official implementations and recommended settings. We additionally include **FLUX.1** and the FLUX.1-based Regional Prompting for direct same-backbone comparison(Chen et al., 2024a).

**Metrics.** Following recent VLM-based evaluation protocols for compositional generation (Hu et al., 2023; Li et al., 2024; Yang et al., 2026), we evaluate prompt faithfulness along three dimensions: $\mathrm{Acc}_{\mathrm{exist}}$, $\mathrm{Acc}_{\mathrm{attribute}}$, and $\mathrm{Acc}_{\mathrm{relationship}}$, measuring element presence, attribute binding, and relationship correctness, respectively. We additionally report CLIPScore (Hessel et al., 2021), HPSv2 (Wu et al., 2023b), and PickScore (Kirstain et al., 2023) to evaluate global text–image alignment and perceptual quality.

Table 1: Quantitative comparison on T2I-CompBench.

| Method | $\mathrm{Acc}_{\mathrm{exist}}$ ↑ | $\mathrm{Acc}_{\mathrm{attribute}}$ ↑ | $\mathrm{Acc}_{\mathrm{relationship}}$ ↑ | CLIPScore ↑ | HPSv2 ↑ | PickScore ↑ |
|---|---|---|---|---|---|---|
| FLUX.1 | 0.79 | 0.66 | 0.60 | 27.15 | 0.29 | 22.89 |
| GLIGEN | 0.67 | 0.43 | 0.62 | 25.74 | 0.25 | 21.40 |
| Composable | 0.61 | 0.41 | 0.44 | 24.77 | 0.23 | 20.86 |
| Attend-and-Excitep | 0.64 | 0.42 | 0.28 | 26.39 | 0.25 | 21.31 |
| MS-Diffusion | 0.81 | 0.51 | 0.53 | 27.01 | 0.28 | 22.31 |
| LLM-Blueprint | 0.78 | 0.50 | 0.64 | 26.34 | 0.26 | 21.54 |
| RPG | 0.80 | 0.55 | 0.70 | 25.95 | 0.29 | 22.59 |
| regional-prompt | 0.88 | 0.69 | 0.73 | 30.30 | 0.29 | 22.82 |
| Graph2Scene | **0.94** | **0.80** | **0.89** | **32.10** | **0.31** | **23.12** |




Figure 5: Visualization examples on simple scenes generated by different control methods.

Table 2: Quantitative comparison on ConceptMix.

| Method | $\mathrm{Acc}_{\mathrm{exist}}$ ↑ | $\mathrm{Acc}_{\mathrm{attribute}}$ ↑ | $\mathrm{Acc}_{\mathrm{relationship}}$ ↑ | CLIPScore ↑ | HPSv2 ↑ | PickScore ↑ |
|---|---|---|---|---|---|---|
| FLUX.1 | 0.75 | 0.31 | 0.42 | 24.82 | 0.27 | 22.31 |
| GLIGEN | 0.28 | 0.04 | 0.06 | 19.28 | 0.21 | 20.04 |
| Composable | 0.17 | 0.17 | 0.04 | 16.51 | 0.19 | 19.44 |
| Attend-and-Excitep | 0.33 | 0.06 | 0.10 | 20.20 | 0.22 | 20.50 |
| MS-Diffusion | 0.47 | 0.06 | 0.11 | 21.76 | 0.28 | 21.97 |
| LLM-Blueprint | 0.29 | 0.06 | 0.11 | 19.66 | 0.21 | 20.09 |
| RPG | 0.40 | 0.07 | 0.09 | 21.88 | 0.27 | 21.93 |
| regional-prompt | 0.80 | 0.64 | 0.59 | 25.91 | 0.29 | 21.76 |
| Graph2Scene | **0.88** | **0.74** | **0.84** | **26.45** | **0.29** | **22.45** |

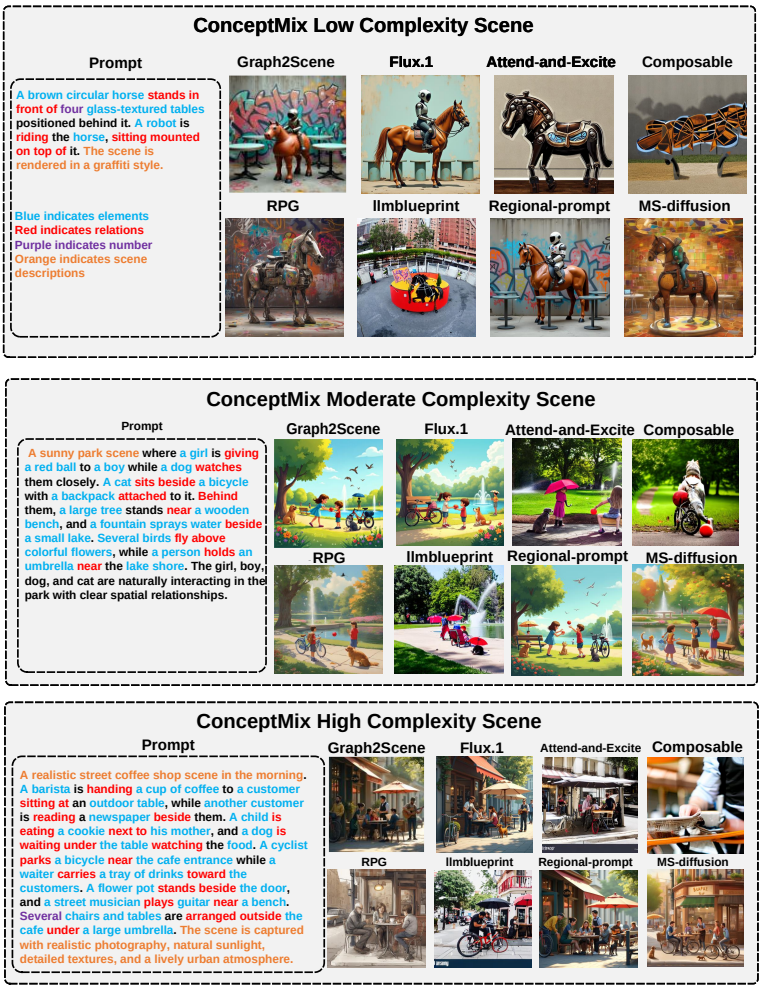



Figure 6: Visualization examples on ConceptMix with increasing scene complexity.

### 4.2 Comparison on Simple Scenes

We first evaluate Graph2Scene on T2I-CompBench, which targets relatively simple scenes: object existence, attribute binding, and spatial composition. Existing methods improve these aspects by editing the prompt or adding spatial conditions. However, they mostly ground objects at the object level and leave the visual interactions to the diffusion model. When several objects compete for the same attention, attributes still leak between them and object counts still go wrong. Graph2Scene adds graph-guided self-attention, which lets related regions exchange information during denoising. This further improves quality, reaching 0.94, 0.80, and 0.89 on existence, attribute, and relationship accuracy, respectively. Compared with the strongest regional control baseline, regional-prompt, Graph2Scene improves attribute accuracy from 0.69 to 0.80 and relationship accuracy from 0.73 to 0.89. This shows that explicit interaction modeling remains beneficial even for simple scenes.

### 4.3 Comparison on Complex Scenes

We then evaluate Graph2Scene on ConceptMix, where each prompt combines many concepts and interactions. As shown in Table 2, Graph2Scene ranks first on five of the six metrics and ties for the best on the last one. It reaches 0.74 on $\text{Acc}_{\text{attribute}}$ and 0.84 on $\text{Acc}_{\text{relationship}}$. It also gives the best existence accuracy (0.88), and CLIPScore (26.45), and its HPSv2 (0.29) matches the top baseline. The gap is largest on relationships, which is exactly what our method targets.

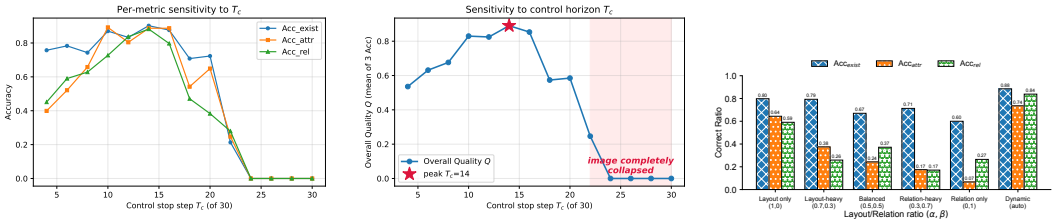
Most baselines perform poorly on complex prompts, partly because several of them use older SD-based backbones, while layout-based methods mainly control where objects appear rather than how they interact. For a fairer comparison, we focus on methods using the same FLUX.1 backbone.

The base FLUX.1 achieves 0.75 on $\text{Acc}_{\text{exist}}$ but only 0.42 on $\text{Acc}_{\text{relationship}}$. Regional Prompting improves attribute accuracy to 0.64 and relationship accuracy to 0.59 by assigning separate prompts to different regions, but many interactions are still missing. RPG shows a similar limitation on a weaker backbone, with only 0.09 relationship accuracy. These results indicate that region-level control helps object placement and attribute binding, but is still insufficient for modeling interactions in complex scenes.

Graph2Scene closes this gap. It does not stop at placing elements. Its graph edges let related regions share information during denoising. This way, the model builds the interaction between elements. On the same FLUX.1 backbone, the relationship score rises from 0.59 to 0.84, and attributes and existence also improve. This shows that the gain comes from relation-level control, not from a stronger base model.

### 4.4 Parameter Sensitivity Analysis

**Effect of the control horizon $T_c$.** The control horizon $T_c$ determines how many denoising steps apply graph control. Figure 7 shows the effect of different control horizons on both individual metrics and overall quality. As shown in Figure. 7b, the overall quality improves from 0.53 at $T_c$=4 to the best score of 0.89 at $T_c$=14, but decreases when the control is applied for too long. Figure 7a further shows that object existence improves quickly with short control (0.76 at $T_c$=4), while attributes and relationships require longer guidance to form. After $T_c$=14, excessive control starts to harm generation. The quality drops to 0.25 at $T_c$=22, and the images collapse when $T_c \geq 24$. This is because later denoising steps mainly refine visual details, where strong control can interfere with the model's natural generation process. A more detailed analysis is provided in Appendix A.2.




(a) Per-metric sensitivity to control horizon $T_c$.

(b) Overall-quality sensitivity to control horizon $T_c$.

(c) Sensitivity to the layout–relation control ratio.

Figure 7: Parameter sensitivity analysis of Graph2Scene.

**Effect of the layout–relation ratio $(\alpha, \beta)$.** We next fix the two control strengths to different values and compare them with our dynamic schedule. As shown in Figure 7c, no fixed ratio works well. Using layout alone $(1, 0)$ already gives 0.64 on attributes and 0.59 on relationships. Surprisingly, any fixed mix does worse. The balanced setting $(0.5, 0.5)$ drops to 0.24 and 0.37, and the relation-only setting $(0, 1)$ drops to 0.07 and 0.27. The reason is timing. A fixed $\beta$ applies relation control from the very first step, before the layout is settled. The two controls then fight each other, and both attributes and relationships suffer. Our dynamic schedule avoids this. It lets layout form first, adds relation control later, and then releases both. This raises all three metrics to the highest values (0.88, 0.74, 0.84), and matches how diffusion models build an image from layout to detail.

## 5 Conclusion

In this paper, we present Graph2Scene, a training-free framework that compiles scene graphs into attention control for compositional text-to-image generation. Based on the observation that existing attention-control methods mainly focus on individual element placement while overlooking inter-object relations, Graph2Scene explicitly models both elements and their interactions through graph-guided attention. Our framework converts scene graph structures into layout- and relation-aware attention constraints, and applies them at different denoising stages with a complexity-aware control schedule. Graph2Scene is compatible with existing diffusion models without additional training and can be directly applied to complex prompts with multiple interacting elements. Extensive experiments demonstrate that Graph2Scene improves compositional generation quality, especially for complex scenes with dense interactions.

## 6 AI DISCLOSURE

Generative AI tools were used to check grammatical errors in the manuscript. In addition, our experiments use the ConceptMix benchmark, which requires specifying the complexity level $K$ and constructing prompts with different numbers of compositional concepts, following the evaluation protocol of the original work (Wu et al., 2024). Generative AI was used to assist in constructing some of these benchmark prompts under the predefined $K$ settings and evaluation rules. All generated prompts were manually reviewed to ensure that they satisfied the benchmark requirements. We take responsibility for the final content of this work, including the manuscript text and experimental prompts produced with the assistance of generative AI.

## REFERENCES

BlackForestLabs. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space, 2025. URL https://arxiv.org/abs/2506.15742.

Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1209–1218. IEEE, 2018.

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.

Anthony Chen, Jianjin Xu, Wenzhao Zheng, Gaole Dai, Yida Wang, Renrui Zhang, Haofan Wang, and Shanghang Zhang. Training-free regional prompting for diffusion transformers. *arXiv preprint arXiv:2411.02395*, 2024a.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-$\alpha$: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *International Conference on Learning Representations*, volume 2024, pp. 57611–57640, 2024b.

Pengtao Chen, Mingzhu Shen, Peng Ye, Jianjian Cao, Chongjun Tu, Christos-Savvas Bouganis, Yiren Zhao, and Tao Chen. $\Delta$-DiT: A Training-Free Acceleration Method Tailored for Diffusion Transformers. *arXiv preprint arXiv:2406.01125*, 2024c.

Zhennan Chen, Yajie Li, Haofan Wang, Zhibo Chen, Zhengkai Jiang, Jun Li, Qian Wang, Jian Yang, and Ying Tai. Ragd: Regional-aware diffusion model for text-to-image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19331–19341, October 2025.

Omer Dahary, Or Patashnik, Kfir Aberman, and Daniel Cohen-Or. Be yourself: Bounded attention for multi-subject text-to-image generation. In *European Conference on Computer Vision*, pp. 432–448. Springer, 2024.

Hanan Gani, Shariq Farooq Bhat, Muzammal Naseer, Salman H. Khan, and Peter Wonka. LLM blueprint: Enabling text-to-image generation with complex and detailed prompts. In *The Twelfth International Conference on Learning Representations*, 2024.

Roei Herzig, Amir Bar, Huijuan Xu, Gal Chechik, Trevor Darrell, and Amir Globerson. Learning canonical representations for scene graph to image generation. In *European Conference on Computer Vision*, pp. 210–227. Springer, 2020.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 7514–7528, 2021.

Jiun Tian Hoe, Xudong Jiang, Chee Seng Chan, Yap-Peng Tan, and Weipeng Hu. Interactdiffusion: Interaction control in text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6180–6189, June 2024a.

Jiun Tian Hoe, Xudong Jiang, Chee Seng Chan, Yap-Peng Tan, and Weipeng Hu. Interactdiffusion: Interaction control in text-to-image diffusion models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6180–6189. IEEE, 2024b.

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. In *2023 ieee/cvf international conference on computer vision (iccv)*, pp. 20349–20360. IEEE, 2023.

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*, 36:78723–78747, 2023.

Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1219–1228. IEEE, 2018.

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. *Advances in neural information processing systems*, 36:36652–36663, 2023.

Haeil Lee, Hansang Lee, Seoyeon Gye, and Junmo Kim. Beta sampling is all you need: Efficient image generation strategy for diffusion models using stepwise spectral analysis. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 4215–4224. IEEE, 2025.

Baiqi Li, Zhiqiu Lin, Deepak Pathak, Jiayao Li, Yixin Fei, Kewen Wu, Tiffany Ling, Xide Xia, Pengchuan Zhang, Graham Neubig, et al. Genai-bench: Evaluating and improving compositional text-to-visual generation. *arXiv preprint arXiv:2406.13743*, 2024.

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 *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22511–22521. IEEE, 2023.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

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*, volume 38, pp. 3657–3665, 2024.

Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European conference on computer vision*, pp. 423–439. Springer, 2022.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. Deepcache: Accelerating diffusion models for free. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15762–15772. IEEE, 2024.

Arash Marioriyad, Mohammadali Banayeeanzade, Reza Abbasi, Mohammad Hossein Rohban, and Mahdieh Soleymani Baghshah. Attention overlap is responsible for the entity missing problem in text-to-image diffusion models! *arXiv preprint arXiv:2410.20972*, 2024.

Shibin Mei, Hang Wang, and Bingbing Ni. Semantic derivative flow: Graph-guided diffusion for controllable instance interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14822–14831, 2026.

Zizheng Pan, Bohan Zhuang, De-An Huang, Weili Nie, Zhiding Yu, Chaowei Xiao, Jianfei Cai, et al. T-stitch: Accelerating sampling in pre-trained diffusion models with trajectory stitching. In *International Conference on Learning Representations*, volume 2025, pp. 6103–6137, 2025.

Royi Rassin, Eran Hirsch, Daniel Glickman, Shauli Ravfogel, Yoav Goldberg, and Gal Chechik. Linguistic binding in diffusion models: Enhancing attribute correspondence through attention map alignment. *Advances in Neural Information Processing Systems*, 36:3536–3559, 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10674–10685. ieee, 2022.

Guibao Shen, Luozhou Wang, Jiantao Lin, Wenhang Ge, Chaozhe Zhang, Xin Tao, Di Zhang, Pengfei Wan, Guangyong Chen, Yijun Li, et al. Scene graph guided generation: Enable accurate relations generation in text-to-image models via textural rectification. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15437–15446. IEEE, 2025.

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*, volume 38, pp. 5544–5552, 2024.

Xierui Wang, Siming Fu, Qihan Huang, Wanggui He, and Hao Jiang. Ms-diffusion: Multi-subject zero-shot image personalization with layout guidance. In *International Conference on Learning Representations*, volume 2025, pp. 95118–95146, 2025.

Qiucheng Wu, Yujian Liu, Handong Zhao, Trung Bui, Zhe Lin, Yang Zhang, and Shiyu Chang. Harnessing the spatial-temporal attention of diffusion models for high-fidelity text-to-image synthesis. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7732–7742. IEEE, 2023a.

Xiaoshi Wu, Keqiang Sun, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score: Better aligning text-to-image models with human preference. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2096–2105. IEEE, 2023b.

Xindi Wu, Dingli Yu, Yangsibo Huang, Olga Russakovsky, and Sanjeev Arora. Conceptmix: A compositional image generation benchmark with controllable difficulty. *Advances in Neural Information Processing Systems*, 37:86004–86047, 2024.

Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. Boxdiff: Text-to-image synthesis with training-free box-constrained diffusion. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7418–7427. IEEE, 2023.

Hongji Yang, Yucheng Zhou, Wencheng Han, Runzhou Tao, Zhongying Qiu, Jianfei Yang, and Jianbing Shen. Hicogen: Hierarchical compositional text-to-image generation in diffusion models via reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 36793–36802, 2026.

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.

Ling Yang, Zhaochen Yu, Chenlin Meng, Minkai Xu, Stefano Ermon, and Bin Cui. Mastering text-to-image diffusion: Recaptioning, planning, and generating with multimodal llms. In *Forty-first International Conference on Machine Learning*, 2024.

Hui Zhang, Tingwei Gao, Jie Shao, and Zuxuan Wu. Blockdance: Reuse structurally similar spatio-temporal features to accelerate diffusion transformers. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12891–12900. IEEE, 2025.

Jiewei Zhang, Song Guo, Peiran Dong, Jie Zhang, Ziming Liu, Yue Yu, and Xiao-Ming Wu. Easing concept bleeding in diffusion via entity localization and anchoring. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 58521–58534. PMLR, 2024.

Yucheng Zhou, Jiahao Yuan, and Qianning Wang. Draw all your imagine: A holistic benchmark and agent framework for complex instruction-based image generation. *arXiv preprint arXiv:2505.24787*, 2025.

# A Appendix

## A.1 A.1Graph Construction Details

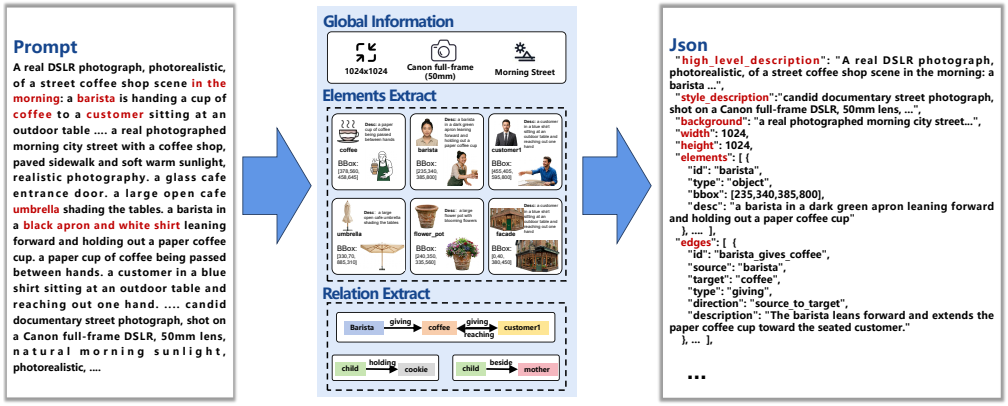



Figure 8: Prompt-to-JSON conversion template used in Graph2Scene.

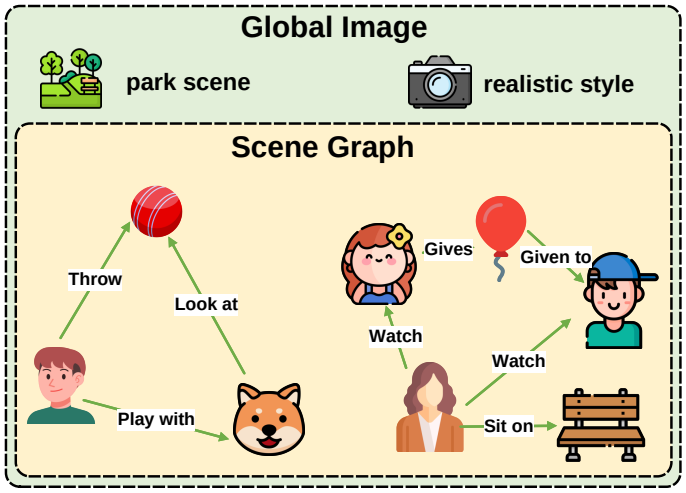



Figure 9: Organization of image information in Graph2Scene.

**Graph construction.** Graph2Scene constructs the scene graph directly from the input prompt using a VLM. As illustrated in Figure. 8, we prompt the VLM to organize the scene information into three parts: global information describing the overall scene and style, individual elements with their descriptions and predicted bounding boxes, and relations specifying how pairs of elements interact. The extracted information is returned in a structured JSON format, where elements correspond to graph nodes and relations correspond to graph edges.

**Node properties.** Each element node contains four basic properties. `id` uniquely identifies the element, while `type` records its semantic category. `bbox` specifies the predicted spatial region of the element in the image, and `desc` provides its local visual description, including appearance and attributes. These properties provide the semantic and spatial information required to associate each element with its corresponding text and image tokens.

**Edge properties.** Each relation edge describes an interaction between two element nodes. `source` and `target` specify the participating elements, `type` records the relation category, and `direction` indicates its direction when applicable. `description` provides a natural-language description of the relation, which is used to identify the corresponding relation tokens for subsequent attention control. Each edge also has a unique `id` for reference.

**Style and global organization.** In addition to elements and relations, we separately organize scene-level information that applies to the entire image. This information includes the background, overall

visual style, lighting, viewpoint, and other imaging conditions. Such global descriptions are shared across all elements and relations rather than being associated with a specific node or edge.

**Prompt organization.** During graph construction, the VLM also organizes the original prompt according to the extracted structure. Scene-level descriptions are retained as global information, while element and relation descriptions are associated with their corresponding nodes and edges. The resulting structured prompt is used as the textual condition for generation, allowing Graph2Scene to identify the text tokens associated with each element and relation during attention control.

### A.2 Analysis of Graph Control Duration

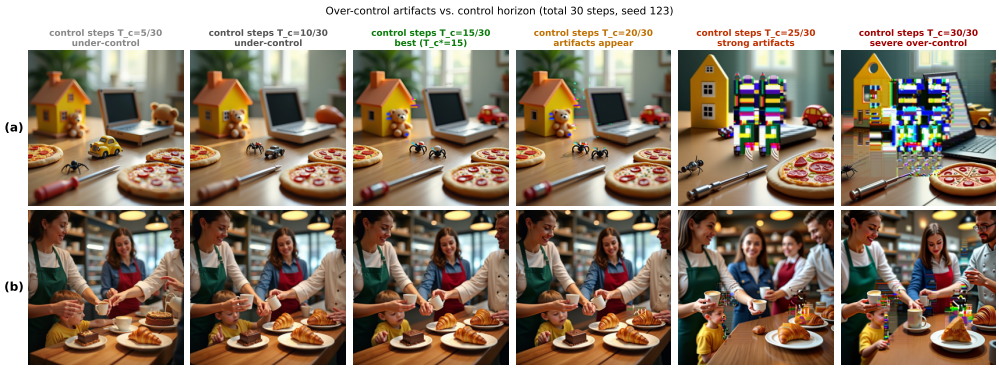



Figure 10: Examples of artifacts caused by excessive control.

Modern diffusion Transformers are pretrained with full attention, where each token can interact with all other tokens during denoising. Graph2Scene modifies this process at inference time by restricting some interactions according to the scene graph. Such control helps the model better establish object layouts and relations. However, the pretrained model is not optimized for this modified attention pattern throughout the entire denoising process. If the control is applied for too many steps, the model increasingly deviates from its original full-attention behavior and cannot properly adapt to the modified attention computation Chen et al. (2024a); Wu et al. (2023a). As a result, the generation process becomes unstable and may eventually produce severe visual artifacts.

This phenomenon is also consistently observed in our experiments. As shown quantitatively in Figure. 7(a) and (b), generation quality drops sharply when the control is extended beyond about 20 denoising steps. When the control horizon reaches 25 steps or more, many generated images are close to complete collapse and no longer contain recognizable visual content. The qualitative examples in Figure. 10 further illustrate this behavior. Slight artifacts already appear around step 20, and become increasingly severe as the control duration increases. These results show that, under the attention modification used by Graph2Scene, longer control does not continuously improve generation. Instead, excessive control increasingly interferes with the pretrained model's original full-attention generation process and eventually leads to unstable generation or visual collapse.

Based on these observations and experimental results, we set the maximum control horizon to $T_c = 0.6T$.

### A.3 Additional Comparison with Scene Graph-Based Methods

Scene graph-based generation methods aim to address the same fundamental challenge as Graph2Scene: generating images with accurate objects, attributes, and relational interactions from structured descriptions. Therefore, we further compare Graph2Scene with representative scene graph-based approaches, including **SG2IM** Johnson et al. (2018), **SGDiff** Yang et al. (2022) , and **CanonicalSg2Im** Herzig et al. (2020), to evaluate the effectiveness of explicitly executing scene graph constraints during the generation process.

**Experiment Setting.** All compared scene graph-based methods provide publicly available models trained on the COCO datasetLin et al. (2014). We directly use their released COCO checkpoints for inference and evaluate all methods on the COCO-Stuff datasetCaesar et al. (2018) with an NVIDIA

Table 3: Quantitative comparison with scene graph-based methods.

| Method | $Acc_{exist}$ ↑ | $Acc_{attribute}$ ↑ | $Acc_{relationship}$ ↑ | CLIPScore ↑ | HPSv2 ↑ | PickScore ↑ |
|---|---|---|---|---|---|---|
| SG2IM | 0.553 | 0.286 | 0.366 | 16.552 | 0.125 | 17.976 |
| SGDiff | 0.336 | 0.208 | 0.253 | 19.307 | 0.131 | 18.453 |
| CanonicalSg2Im | 0.032 | 0.016 | 0.000 | 12.195 | 0.085 | 16.349 |
| Graph2Scene | **0.927** | **0.893** | **0.872** | **26.296** | **0.294** | **22.845** |

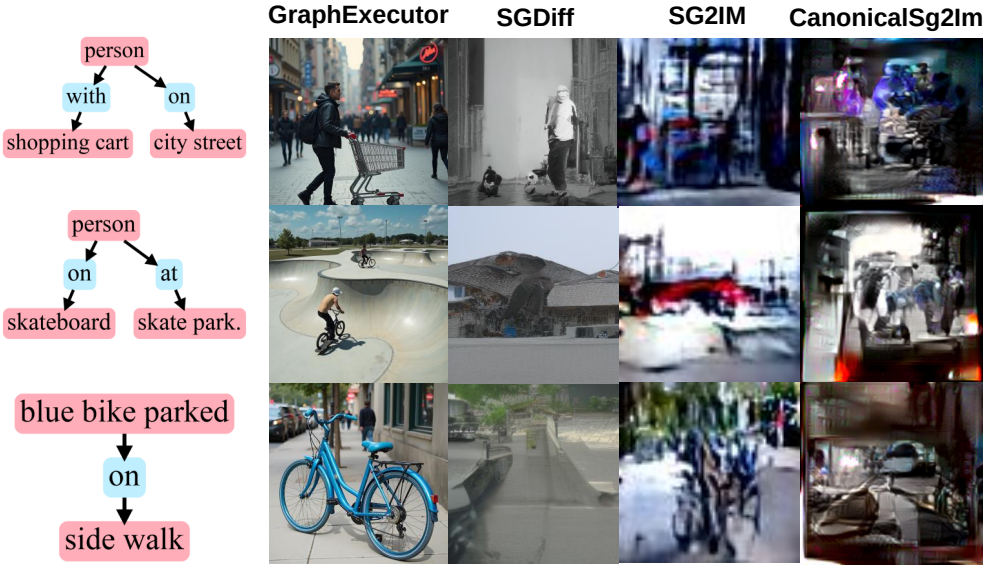



Figure 11: Case studies comparing Graph2Scene with existing Scene Graph based methods on the COCO-Stuff dataset.

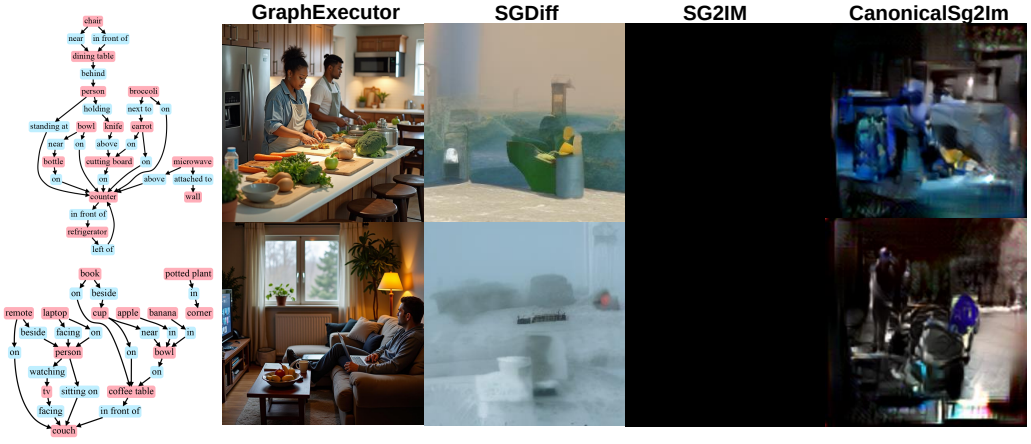



Figure 12: Case studies comparing Graph2Scene with existing scene graph-based methods on the open complex prompts.

L20 GPU under the same experimental setting. Following our main experiments, we adopt the same evaluation metrics, including object existence accuracy ($\mathrm{Acc}_{\mathrm{exist}}$), attribute accuracy ($\mathrm{Acc}_{\mathrm{attribute}}$), relationship accuracy ($\mathrm{Acc}_{\mathrm{relationship}}$), CLIPScore, HPSv2, and PickScore, to measure both semantic alignment and overall generation quality.

**Quantitative Comparison.** Table 3 shows that Graph2Scene consistently outperforms existing scene graph-based methods across all metrics. SG2IM uses graph convolutions to predict layouts and a GAN-based refinement network for generation. SGDiff employs masked contrastive pretraining to encode scene graphs into a latent diffusion model. CanonicalSg2Im learns canonical graph representations to handle semantically equivalent graphs. However, these methods struggle with visual fidelity due to limited generation capacity and imperfect alignment between graph structures and visual content. In contrast, Graph2Scene leverages a powerful pretrained text-to-image diffusion backbone and explicitly compiles scene graph structures into attention operations. This design integrates structural constraints while preserving the model's native generation ability. As a result, Graph2Scene achieves substantial improvements in object existence, attribute recognition, and relationship accuracy, reaching scores of 0.927, 0.893, and 0.872, respectively. These results demonstrate more effective preservation of scene semantics and interactions compared with previous scene graph-based approaches. Moreover, Graph2Scene obtains higher overall visual quality scores, achieving a CLIPScore of 26.296, an HPSv2 score of 0.294, and a PickScore of 22.845, indicating better alignment with both image-text similarity and human preference.

**Case Study on COCO-Stuff.** Figure 11 presents qualitative comparisons on the COCO-Stuff dataset. Although all methods receive the same scene graph conditions, Graph2Scene produces significantly clearer and more realistic images while preserving the specified objects and relationships. This improvement is mainly attributed to the stronger generative capability of modern diffusion backbones and the explicit execution of scene graph constraints during generation. In contrast, previous scene graph-based methods are built upon earlier generation architectures, such as GAN-based pipelines or early scene graph-conditioned diffusion models, which often suffer from limited visual fidelity and difficulty in aligning graph structures with image contents. As shown in Figure 11, SGDiff can generate some recognizable scene elements, such as the person and skateboard, but many details remain blurry and incomplete. For SG2IM and CanonicalSg2Im, although the generated images contain rough color distributions and coarse object regions, the objects are often heavily distorted, making their categories and attributes difficult to identify. For example, the shopping cart, city street, and bicycle structures are only weakly represented, and the generated contents lack clear boundaries and realistic appearances.

**Case Study on Complex Open Prompts.** Figure 12 further presents qualitative comparisons on open complex prompts with larger and more diverse scene graphs. Compared with the COCO-Stuff examples, these prompts contain more objects, attributes, and relational constraints, posing greater challenges for scene graph-based generation. Graph2Scene consistently generates clear and coherent images while preserving multiple objects and their interactions, demonstrating its ability to generalize to complex scene structures. In contrast, previous methods suffer from severe degradation when handling such complex inputs. SGDiff can still generate some rough scene layouts, but the generated images remain highly blurred and fail to recover detailed object appearances. For SG2IM, the model completely collapses and produces nearly black images under these open complex prompts. This failure is mainly because SG2IM relies on a graph encoder and layout prediction pipeline learned from fixed training distributions, making it difficult to handle unseen graph compositions and substantially larger scene structures.

### A.4 Prompt and Evaluation Templates

This section provides the prompt templates used in Graph2Scene. We include the VLM prompt used to construct the structured scene graph from the input text, together with the templates used for VLM-based evaluation of object existence, attribute correctness, and relationship accuracy. The complete prompt formats are shown in the following figures.

**VLM Graph Construction Prompt**

**Role.** You are a scene-layout planner for graph-conditioned image generation.
**Task.** Decompose an image prompt into a scene graph of objects and typed relations, with an absolute pixel bounding box for every object.
**Output. Return STRICT JSON only. No markdown, code fences, or explanation.**

**Schema**

```
{
  "high_level_description": "one sentence describing the whole scene",
  "background": "the setting / environment only",
  "elements": [
    {"id": "snake_case_unique_id", "type": "object",
     "bbox": [x1, y1, x2, y2],
     "desc": "concrete visual description of this single object"}
  ],
  "edges": [
    {"id": "snake_case_edge_id", "source": "element_id",
     "target": "element_id", "type": "one of allowed_edge_types",
     "relation_phrase": "exact word(s) from the Prompt",
     "strength": 0.0-1.0,
     "direction": "source_to_target | bidirectional",
     "phase": "early | mid | late"}
  ]
}
```

**Rules**

**Canvas:** Every bbox must lie within the given image width and height.
**Elements:** Use at most {max_nodes} elements; keep foreground subjects reasonably large.
**Layout:** Avoid heavy bbox overlap unless the relation requires it (e.g., on / holding).
**Edges:** Create an edge only when the prompt implies a real interaction or spatial relation.
**Descriptions:** Keep each element description concise and unambiguous.
**Edge type:** Use the most specific interaction verb supported by the prompt.
**Relation phrase:** Copy the actual relation word(s) from the prompt verbatim; use an empty string only when the relation is implicit.

**Input**

**Image size:** width={width}, height={height}
**Prompt:** {prompt}
**Optional grounding: if a reference image is provided, ground the bounding boxes on the actual object positions.**
**Instruction: Return the scene-graph JSON now.**

Figure 13: Prompt template used for scene graph construction.

**Acc_exist — Object Existence Accuracy**

**Task:** Check whether the specified object appears in the image.
**Original prompt:** "{original_prompt}" **Object:** "{object}"
**Instruction:** Analyze the image carefully and determine whether the specified object is clearly present and recognizable.
**Decision rules: Yes — the object is visible.**
**No — the object is missing or replaced by another object. Do not infer its presence from the prompt.**
**Unclear — use only when the visual evidence is insufficient.**
**Output only: Description:** [Brief image description] **Reason:** [Brief reason] **Output: [Yes / No / Unclear]**

**Acc_attribute — Attribute Accuracy**

**Task:** Check whether the specified object matches the attributes described in the text.
**Original prompt:** "{original_prompt}" **Object description:** "{attribute_description}"
**Instruction:** Analyze the image carefully. Determine whether the object is present and whether its visible attributes—such as color, shape, material, pattern, size, count, or structure—match the description.
**Decision rules: Yes — the object is visible and its attributes match.**
**No — the object is missing, an attribute is incorrect, or the attributes are assigned to another object.**
**Unclear — use only when the visual evidence is insufficient.**
**Output only: Description:** [Brief image description] **Reason:** [Brief reason] **Output: [Yes / No / Unclear]**

**Acc_relationship — Relationship Accuracy**

**Task:** Check whether the specified relationship is correctly shown in the image.
**Original prompt:** "{original_prompt}" **Relationship:** "{relationship_description}"
**Instruction:** Analyze the image carefully. Determine whether all involved objects are visible and whether the described relationship is correctly presented between the intended objects.
**Decision rules: Yes — the relationship is clearly shown between the correct objects.**
**No — an object is missing, the relationship is incorrect, the direction is reversed, or the interaction involves the wrong object.**
**Unclear — use only when the visual evidence is insufficient.**
**Output only: Description:** [Brief image description] **Reason:** [Brief reason] **Output: [Yes / No / Unclear]**

Figure 14: Prompt templates used to evaluate $\mathrm{Acc}_{\mathrm{exist}}$, $\mathrm{Acc}_{\mathrm{attribute}}$, and $\mathrm{Acc}_{\mathrm{relationship}}$ with a VLM.

### A.5 Implementation Details of the Graph Execution Scheduler

We extract element tokens and relation tokens from the prompt, denoted by $\mathcal{U}_{\mathrm{elem}}$ and $\mathcal{U}_{\mathrm{rel}}$, respectively. For each token $u$, we collect its cross-attention maps over all denoising steps and aggregate them into a single attention map $A_u$.

For each token $u$, we obtain a set of high-response regions and their normalized attention scores, denoted by

$$\mathcal{C}_u = \{c_u^1, c_u^2, \ldots\}, \tag{4}$$

where each $c \in \mathcal{C}_u$ denotes the attention score of a high-response region.

**Attention leakage.** We regard the region with the highest attention score as the dominant response of token $u$, and treat the remaining response as attention leakage:

$$L_u = 1 - \max_{c \in \mathcal{C}_u} c. \tag{5}$$

**Attention overlap.** Attention overlap measures the spatial confusion between concepts of the same type. We compute overlap only between element tokens or between relation tokens, since relation attention can naturally overlap with its participating elements. Let $\mathcal{H}_u$ denote the high-response spatial region of token $u$. For $u \neq v$, where either $u, v \in \mathcal{U}_{\text{elem}}$ or $u, v \in \mathcal{U}_{\text{rel}}$, we define

$$O_{u,v} = \frac{|\mathcal{H}_u \cap \mathcal{H}_v|}{|\mathcal{H}_u \cup \mathcal{H}_v|}. \tag{6}$$

After computing the two attention statistics, we take the largest leakage value over all element and relation tokens as $L_{\max}$, and the largest overlap value among element-element and relation-relation token pairs as $O_{\max}$.

We use the more severe attention failure to determine the control ratio and the corresponding control duration:

$$\rho = \max(L_{\max}, O_{\max}), \qquad T_{\text{ctrl}} = \rho T_c, \tag{7}$$

where $\rho \in [0, 1]$ represents the required control level and $T_c$ denotes the maximum control duration, whose setting is discussed in Appendix A.2.

### A.6 Additional Qualitative Results

We further evaluate Graph2Scene on the more challenging LongBench-T2I benchmark, which is designed to assess text-to-image models under long and highly compositional prompts containing many objects, attributes, and relations. Compared with conventional benchmarks using relatively short descriptions, LongBench-T2I places greater demands on the model's ability to preserve multiple visual elements and correctly organize their interactions within a single scene.

As shown in Figure. 15, we further use a VLM to evaluate the generated images in terms of object existence, attribute correctness, and relationship accuracy. Under these extremely long and complex prompts, all methods experience noticeable degradation: some required objects are omitted, attributes are incorrectly assigned or lost, and many relationships fail to appear in the generated scenes. This highlights the difficulty of faithfully following LongBench-T2I prompts with a large number of visual concepts and interactions. Nevertheless, Graph2Scene consistently preserves a larger proportion of the specified elements, attributes, and relationships than the compared methods. In particular, its advantage is more evident for attributes and relationships, indicating that explicitly organizing and controlling the interactions among scene elements helps maintain compositional consistency even when the prompt becomes highly complex.

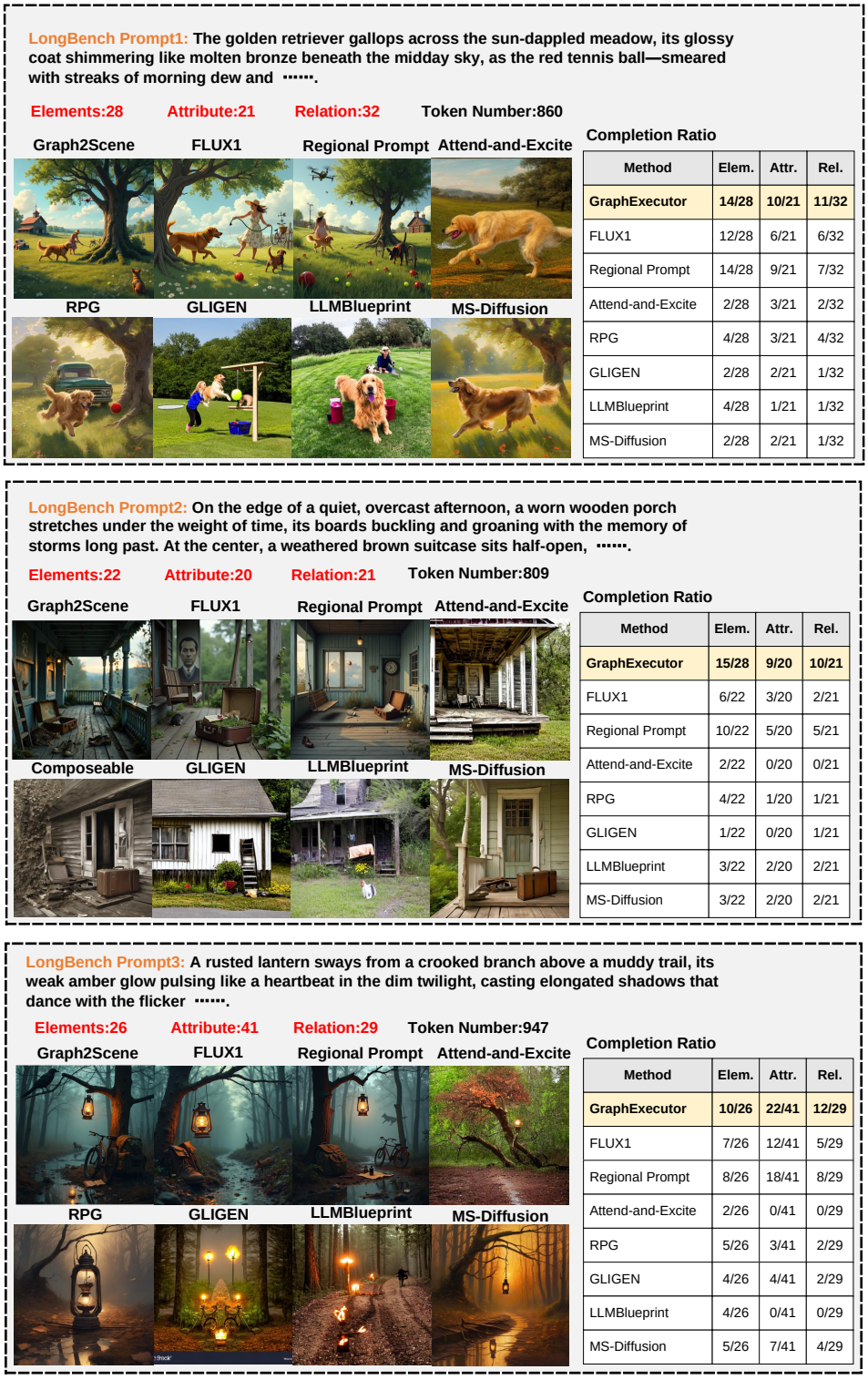



Figure 15: Qualitative comparison on LongBench-T2I.