3DIS: DEPTH-DRIVEN DECOUPLED INSTANCE SYNTHESIS FOR TEXT-TO-IMAGE GENERATION

Anonymous authors
Paper under double-blind review

Figure 1: **Images generated using our 3DIS.** Based on the user-provided layout, 3DIS generates a scene depth map that precisely positions each instance and renders their fine-grained attributes without the need for additional training, using a variety of foundational models.

ABSTRACT

The increasing demand for controllable outputs in text-to-image generation has spurred advancements in multi-instance generation (MIG), allowing users to define both instance layouts and attributes. However, unlike image-conditional generation methods such as ControlNet, MIG techniques have not been widely adopted in state-of-the-art models like SD2 and SDXL, primarily due to the challenge of building robust renderers that simultaneously handle instance positioning and attribute rendering. In this paper, we introduce **D**epth-**D**riven **D**ecoupled **I**nstance **S**ynthesis (3DIS), a novel framework that decouples the MIG process into two stages: (i) generating a coarse scene depth map for accurate instance positioning and scene composition, and (ii) rendering fine-grained attributes using pre-trained ControlNet on any foundational model, without additional training. Our 3DIS framework integrates a custom adapter into LDM3D for precise depth-based layouts and employs a finetuning-free method for enhanced instance-level attribute rendering. Extensive experiments on COCO-Position and COCO-MIG benchmarks demonstrate that 3DIS significantly outperforms existing methods in both layout precision and attribute rendering. Notably, 3DIS offers seamless compatibility with diverse foundational models, providing a robust, adaptable solution for advanced multi-instance generation.

1 INTRODUCTION

With the rapid advancement of text-to-image generation technologies, there is a growing interest in achieving more controllable outputs, which are now widely utilized in artistic creation (Wang et al., 2024; Li et al., 2024a): *(i) Image-conditional generation techniques*, *e.g.*, ControlNet (Zhang et al., 2023), allow users to generate images based on inputs like depth maps or sketches. *(ii) Multi-instance generation (MIG) methods*, *e.g.*, GLIGEN (Li et al., 2023b) and MIGC (Zhou et al., 2024), enable users to define layouts and detailed attributes for each instance within the generated images.

However, despite the importance of MIG in controllable generation, these methods have not been widely adopted across popular foundational models like SD2 (Rombach et al., 2023) and SDXL (Podell et al., 2023), unlike the more widely integrated ControlNet. Current state-of-the-art MIG methods mainly rely on the less capable SD1.5 (Rombach et al., 2022) model.

We argue that the limited adoption of MIG methods is not merely due to ***resource constraints*** but also stems from a more fundamental challenge, *i.e.*, ***unified adapter challenge***. Current MIG approaches train a single adapter to handle both instance positioning and attribute rendering. This unified structure complicates the development of robust renderers for fine-grained attribute details, as it requires large amounts of high-quality instance-level annotations. These detailed annotations are more challenging to collect compared to the types of controls used in image-conditional generation, such as depth maps or sketches.

To address the unified adapter challenge and enable the use of a broader range of foundational models for MIG, we propose a novel framework called **D**epth-**D**riven **D**ecoupled **I**nstance **S**ynthesis (3DIS). 3DIS tackles this challenge by decoupling the image generation process into two distinct stages, as shown in Fig. 2. ***(i) Generating a coarse scene depth map***: During this stage, the MIG adapter ensures accurate instance positioning, coarse attribute alignment, and overall scene harmony without the complexity of fine attribute rendering. ***(ii) Rendering a fine-grained RGB image***: Based on the generated scene depth map, we design a finetuning-free method that leverages any popular foundational model with pretrained ControlNet to guide the overall image generation, focusing on detailed instance rendering. This approach *requires only* ***a single training process*** for the adapter at stage ***(i)***, enabling ***seamless integration with different foundational models*** without needing retraining for each new model.

The 3DIS architecture comprises three key components: ***(i) Scene Depth Map Generation***: We developed the first layout-controllable text-to-depth generation model by integrating a well-designed adapter into LDM3D (Stan et al., 2023). This integration facilitates the generation of precise, depth-informed layouts based on instance conditions. ***(ii) Layout Control***: We introduce a method to leverage pretrained ControlNet for seamless integration of the generated scene depth map into the generative process. By filtering out high-frequency information from ControlNet's feature maps, we enhance the integration of low-frequency global scene semantics, thereby improving the coherence and visual appeal of the generated images. ***(iii) Detail Rendering***: Our method performs Cross-Attention operations separately for each instance to achieve precise rendering of specific attributes (*e.g.*, category, color, texture) while avoiding attribute leakage. Additionally, we use SAM for semantic segmentation on the scene depth map, optimizing instance localization and resolving conflicts from overlapping bounding boxes. This advanced approach significantly improves the rendering of detailed and accurate multi-instance images.

We conducted extensive experiments on two benchmarks to evaluate the performance of 3DIS: *(i) COCO-Position* (Lin et al., 2015; Zhou et al., 2024): Evaluated the layout accuracy and coarse-grained category attributes of the scene depth maps. *(ii) COCO-MIG* (Zhou et al., 2024): Assessed the fine-grained rendering capabilities. The results indicate that 3DIS excels in creating superior scenes while preserving the accuracy of fine-grained attributes during detailed rendering. On the COCO-Position benchmark, 3DIS achieved a **16.3**% improvement in AP_{75} compared to the previous state-of-the-art method, MIGC. On the COCO-MIG benchmark, our training-free detail rendering approach improved the Instance Attribute Success Ratio by **35**% over the training-free method Multi-Diffusion (Bar-Tal et al., 2023) and by **5.5**% over the adapter-based method InstanceDiffusion (Wang et al., 2024). Furthermore, the 3DIS framework can be seamlessly integrated with off-the-shelf adapters like GLIGEN and MIGC, thereby enhancing their rendering capabilities.

In summary, the key contributions of this paper are as follows:

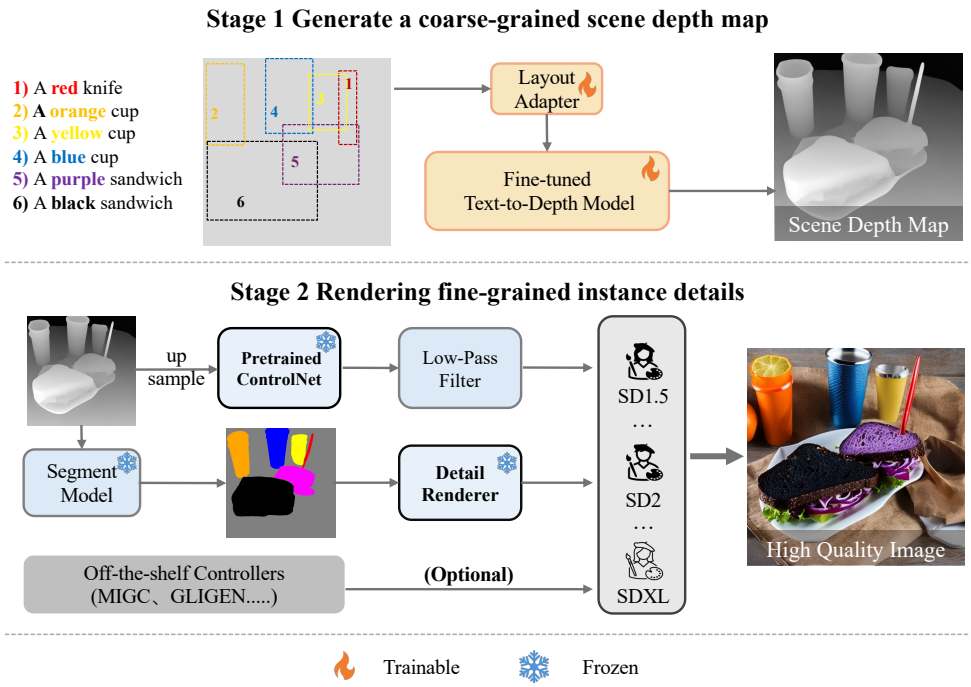


Figure 2: **The overview of 3DIS.** 3DIS decouples image generation into two stages: creating a scene depth map and rendering high-quality RGB images with various generative models. It first trains a Layout-to-Depth model to generate a scene depth map. Then, it uses a pre-trained ControlNet to inject depth information into various generative models, controlling scene representation. Finally, a training-free detail renderer renders the fine-grained attributes of each instance.

- We propose a novel **3DIS framework** that decouples multi-instance generation into two stages: adapter-controlled scene depth map generation and training-free fine-grained attribute rendering, enabling integration with various foundational models.
- We introduce the first **layout-to-depth model** for multi-instance generation, which improves scene composition and instance positioning compared to traditional layout-to-RGB methods.
- Our **training-free detail renderer** enhances fine-grained instance rendering without additional training, significantly outperforming state-of-the-art methods while maintaining compatibility with pretrained models and adapters.

2 Related Work

Controllable Text-to-Image Generation. With the rapid advancements in text-to-image generation technology, current models are capable of producing high-quality images (Rombach et al., 2022; 2023; Podell et al., 2023). Researchers are now increasingly focused on enhancing their control over the generated content. Numerous approaches have been developed to improve this control. ControlNet (Zhang et al., 2023) incorporates user inputs such as depth maps and edge maps by training an additional side network, allowing for precise layout control in image generation. Methods like IPAdapter (Ye et al., 2023) and PhotoMaker (Li et al., 2024b) generate corresponding images based on user-provided portraits. Techniques such as ELITE (Wei et al., 2023) and SSR-Encoder (Zhang et al., 2024) enable networks to accept specific conceptual image inputs for better customization. Additionally, MIGC (Zhou et al., 2024) and InstanceDiffusion (Wang et al., 2024) allow networks to generate images based on user-specified layouts and instance attribute descriptions, defining this task as Multi-Instance Generation (MIG), which is the focal point of this paper.

Multi-Instance Generation (MIG). MIG involves generating each instance based on a given layout and detailed attribute descriptions, while maintaining overall image harmony. Current MIG methods primarily use Stable Diffusion (SD) architectures, classified into three categories: 1) Training-free methods: Techniques like BoxDiffusion (Xie et al., 2023) and RB (Xiao et al., 2023) apply energy

functions to attention maps, enabling zero-shot layout control by converting spatial guidance into gradient inputs. Similarly, Multi-Diffusion (Bar-Tal et al., 2023) generates instances separately and then combines them according to user-defined spatial cues, enhancing control over orientation and arrangement. 2) Adapter methods: Approaches like GLIGEN (Li et al., 2023b) and InstanceDiffusion (Wang et al., 2024) integrate trainable gated self-attention layers into the U-Net (Ronneberger et al., 2015), improving layout assimilation and instance fidelity. MIGC (Zhou et al., 2024) further divides the task, using an enhanced attention mechanism to generate each instance precisely before integration. 3) SD-tuning methods: Reco (Yang et al., 2023) and Ranni (Feng et al., 2024) add instance position data to text inputs and fine-tune both CLIP and U-Net, allowing the network to utilize positional cues for more precise image synthesis. Previous methods entangled instance positioning with attribute rendering, complicating the training of a robust instance renderer. Our approach decouples this process into adapter-controlled scene depth map generation and training-free detail rendering. This separation allows the adapter to only handle instance positioning and coarse attributes, while leveraging the generative priors of pre-trained models, enhancing both flexibility and performance.

3 METHOD

3.1 PRELIMINARIES

Latent Diffusion Models (LDMs) are among the most widely used text-to-image models today. They significantly enhance generation speed by placing the diffusion process for image synthesis within a compressed variational autoencoder (VAE) latent space. To ensure that the generated images align with user-provided text descriptions, LDMs typically employ a Cross Attention mechanism, which integrates textual information into the image features of the network. In mathematical terms, the Cross Attention operation can be expressed as follows:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V}, \tag{1}$$

where $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ represent the query, key, and value matrices derived from the image and text features, respectively, while d_k denotes the dimension of the key vectors.

3.2 OVERVIEW

Fig. 2 illustrates the overview framework of the proposed 3DIS, which decouples image generation into coarse-grained scene construction and fine-grained detail rendering. The specific implementation of 3DIS consists of three steps: **1) Scene Depth Map Generation (§ 3.3)**, which produces a corresponding scene depth map based on the user-provided layout; **2) Global Scene Control (§ 3.4)**, which ensures that the generated images align with the scene maps, guaranteeing that each instance is represented; **3) Detail Rendering (§3.5)**, which ensures that each generated instance adheres to the fine-grained attributes described by the user.

3.3 SCENE DEPTH MAP GENERATION

In this section, we discuss how to generate a corresponding depth map based on the user-provided layout, creating a coherent and well-structured scene while accurately placing each instance.

Choosing the text-to-depth model. Upon investigation, we identified RichDreamer (Qiu et al., 2024) and LDM3D (Stan et al., 2023) as the primary models for text-to-depth generation. RichDreamer fine-tunes the pretrained RGB Stable Diffusion (SD) model to generate 3D information, specifically depth and normal maps, while LDM3D enables SD to produce both RGB images and depth maps simultaneously. Experimental comparisons show LDM3D outperforms RichDreamer in complex scenes, likely due to its concurrent RGB and depth map generation. This dual capability preserves RGB image quality while enhancing depth map generation, making LDM3D our preferred model for text-to-depth generation.

Fine-tuning the text-to-depth model. In contrast to RGB images, depth maps typically prioritize the restoration of low-frequency components over high-frequency details. For instance, while a texture-rich skirt requires intricate details for RGB image generation, its corresponding depth

map remains relatively smooth. Therefore, we aim to enhance the model's ability to recover low-frequency content. Low-frequency components often indicate significant redundancy among adjacent pixels. To simulate this characteristic, we implemented an augmented pyramid noise strategy (Kasiopy, 2023), which involves downsampling and then upsampling randomly sampled noise ϵ to create patterns with high redundancy between adjacent pixels. We used the original SD training loss (Rombach et al., 2022) to fine-tune our text-to-depth model θ, but adjusted the model to predict this patterned noise $\epsilon_{\text{pyramid}}$ with the text prompt $\boldsymbol{c}$:

$$\min_{\theta} \mathcal{L}_{\text{text}} = \mathbb{E}_{z,\epsilon\sim\mathcal{N}(0,I),t}\left[\left\|\epsilon_{\text{pyramid}} - f_{\theta}(z_t, t, c)\right\|_2^2\right]. \tag{2}$$

Training the Layout-to-depth adapter. Similar to previous methodologies (Zhou et al., 2024; Li et al., 2023b; Wang et al., 2024), we incorporated an adapter into our fine-tuned text-to-depth model, enabling layout-to-depth generation, specifically leveraging the state-of-the-art MIGC (Zhou et al., 2024) model. Unlike earlier approaches, our method for generating depth maps does not rely on detailed descriptions of specific instance attributes, such as material or color. Consequently, we have augmented the dataset used for MIGC by eliminating fine-grained attribute descriptions from the instance data, thus focusing more on the structural properties of individual instances and the overall scene composition. The training process for the adapter θ' can be expressed as:

$$\min_{\theta'} \mathcal{L}_{\text{layout}} = \mathbb{E}_{z,\epsilon\sim\mathcal{N}(0,I),t}\left[\left\|\epsilon_{\text{pyramid}} - f_{\theta,\theta'}(z_t, t, c, l)\right\|_2^2\right], \tag{3}$$

where the base text-to-depth model θ is frozen, and the $\boldsymbol{l}$ is the input layout.

3.4 Global Scene Control

In this section, we will describe how to control the generated images to align with the layout of the generated scene depth map, ensuring that each instance appears in its designated position.

Injecting depth maps with ControlNet. After generating scene depth maps with our layout-to-depth models, we employed the widely adopted ControlNet (Zhang et al., 2023) model to incorporate global scene information. Scene depth maps focus on overall scene structure, without requiring fine-grained detail. Thus, although the base model produces 512x512 resolution maps, they can be upsampled to 768x768, 1024x1024, or higher (see Fig. 3 and Fig. 4, e.g., SD2 and SDXL). Since most generative models have depth ControlNet versions, these maps can be applied across various models, ensuring accurate instance placement and mitigating omission issues.

Removing high-frequency noise in depth maps. In our framework, the injected depth maps are designed to manage the low-frequency components of the constructed scene, while the generation of high-frequency details is handled by advanced grounded text-to-image models. To enhance the integration of these components, we implement a filtering process to remove high-frequency noise from the feature maps generated by ControlNet before injecting them into the image generation network. Specifically, the scene condition feature output from ControlNet, denoted as F, is added to the generation network. Prior to this addition, we transform F into the frequency domain via the Fast Fourier Transform (FFT) and apply a filter to attenuate the high-frequency components:

$$F_{\text{filtered}} = \mathcal{F}^{-1}\left(H_{\text{low}} \cdot \mathcal{F}(F)\right), \tag{4}$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the FFT and inverse FFT, respectively, and H_{low} represents a low-pass filter applied in the frequency domain. This approach has been shown to reduce the occurrence of artifacts and improve the overall quality of the generated images without reducing performance.

3.5 Details rendering

Through the control provided by ControlNet, we can ensure that the output images align with our generated scene depth maps, thus guaranteeing that each instance appears at its designated location. However, we still lack assurance regarding the accuracy of attributes such as category, color, and material for each instance. To render each instance with correct attributes, we propose a training-free **detail renderer** to replace the original Cross-Attention Layers for this purpose. The process of rendering an entire scene using a detail renderer can be broken down into the following three steps.

Table 1: **Quantitative results on COCO-Position (§4.3).** We only utilize complex layouts that contain at least five instances, resulting in significant overlap.

	Layout Accuracy			Instance Accuracy			Image Quality	
Method	$AP \uparrow$	$AP_{75} \uparrow$	$AP_{50} \uparrow$	$SR_{inst} \uparrow$	MIoU	$CLIP \uparrow$	$SR_{img} \uparrow$	$FID \downarrow$
BoxDiff [ICCV23]	3.15	2.12	10.92	22.74	27.28	18.82	0.53	25.15
MultiDiff [ICML23]	6.37	4.24	13.22	28.75	34.17	20.12	0.80	33.20
GLIGEN [CVPR23]	38.49	40.75	63.79	83.31	70.14	19.61	40.13	26.80
MIGC [CVPR24]	45.03	46.15	80.09	83.37	71.92	20.07	43.25	24.52
3DIS (SD1.5)	**56.83**	**62.40**	**82.29**	**84.71**	**73.32**	**20.84**	**46.50**	**23.24**
vs. prev. SoTA	**+11.8**	**+16.3**	**+2.2**	**+1.3**	**+1.4**	**+0.8**	**+3.3**	**+1.3**

Rendering each instance separately. For an instance $\boldsymbol{i}$, ControlNet ensures that a shape satisfying its descriptive criteria is positioned within the designated bounding box $\boldsymbol{b}_i$. By applying Cross Attention using the text description of the instance $\boldsymbol{i}$, we can ensure that the attention maps generate significant response values within the $\boldsymbol{b}_i$ region, accurately rendering the attributes aligned with the instance's textual description. For each Cross-Attention layer in the foundation models, we independently render each instance $\boldsymbol{i}$ with their text descriptions to obtain the rendered result $\mathbf{r}_i$, while similarly applying the global image description to yield rendering background $\mathbf{r}_c$. Our next step is to merge the obtained feature maps $\{\mathbf{r}_1, \cdots, \mathbf{r}_n, \mathbf{r}_c\}$ into a single feature map, aligning with the forward pass of the original Cross-Attention layers.

SAM-Enhancing Instance Location. While mering rendering results, acquiring precise instance locations helps prevent attribute leakage between overlapping bounding boxes and maintains structural consistency with the instances in the scene depth maps. Consequently, we employ the SAM (Kirillov et al., 2023) model to ascertain the exact position of each instance. For an instance $\boldsymbol{i}$, by utilizing our generated scene depth map $\mathbf{m}_{scene}$ alongside its corresponding bounding box $\boldsymbol{b}_i$, we can segment the specific shape mask $\mathbf{m}_i$ of this instance, thereby facilitating subsequent merging:

$$\mathbf{m}_i = \text{SAM}(\mathbf{m}_{scene}, \mathbf{b}_i) \tag{5}$$

Merging rendering results. We employ the precise mask $\mathbf{m}_i$ obtained from SAM to constrain the rendering results of instance $\boldsymbol{i}$ to its own region, ensuring no influence on other instances. Specifically, we construct a new mask $\mathbf{m}'_i$ by assigning a value of α to the areas where $\mathbf{m}_i$ equals 1, while setting all other regions to $-\infty$. Simultaneously, we assign a background value of β to the global rendering $\mathbf{r}_c$ through a mask $\mathbf{m}'_c$. By applying the softmax function to the set $\{\mathbf{m}'_1, \mathbf{m}'_2, \ldots, \mathbf{m}'_n, \mathbf{m}'_c\}$, we derive the spatial weights $\{\mathbf{m}''_1, \mathbf{m}''_2, \ldots, \mathbf{m}''_n, \mathbf{m}''_c\}$ for each rendering instance. At each Cross Attention layer, the output can be expressed as follows to render the whole scene:

$$\mathbf{r} = \mathbf{m}''_1 \cdot \mathbf{r}_1 + \mathbf{m}''_2 \cdot \mathbf{r}_2 + \ldots + \mathbf{m}''_n \cdot \mathbf{r}_n + \mathbf{m}''_c \cdot \mathbf{r}_c \tag{6}$$

4 Experiment

4.1 Implement Details

Tuning of text-to-depth models. We utilized a training set comprising 5,878 images from the LAION-art dataset (Schuhmann et al., 2021), selecting only those with a resolution exceeding 512x512 pixels and an aesthetic score of ≥ 8.0. Depth maps for each image were generated using Depth Anything V2 (Yang et al., 2024). Given the substantial noise present in the text descriptions associated with the images in LAION-art, we chose to produce corresponding image captions using BLIP2 (Li et al., 2023a). We employed pyramid noise (Kasiopy, 2023) to fine-tune the LDM3D model for 2,000 steps, utilizing the AdamW (Kingma & Ba, 2017) optimizer with a constant learning rate of $1e^{-4}$, a weight decay of $1e^{-2}$, and a batch size of 320.

Training of the layout-to-depth adapter. We adopted the MIGC (Zhou et al., 2024) architecture as the adapter for layout control. In alignment with this approach, we utilized the COCO dataset (Lin et al., 2015) for training. We employed Stanza (Qi et al., 2020) to extract each instance description from the corresponding text for every image and used Grounding-DINO (Liu et al., 2023) to obtain the image layout. Furthermore, we augmented each instance's description by incorporating modified versions that omitted adjectives, allowing our layout-to-depth adapter to focus more on global scene

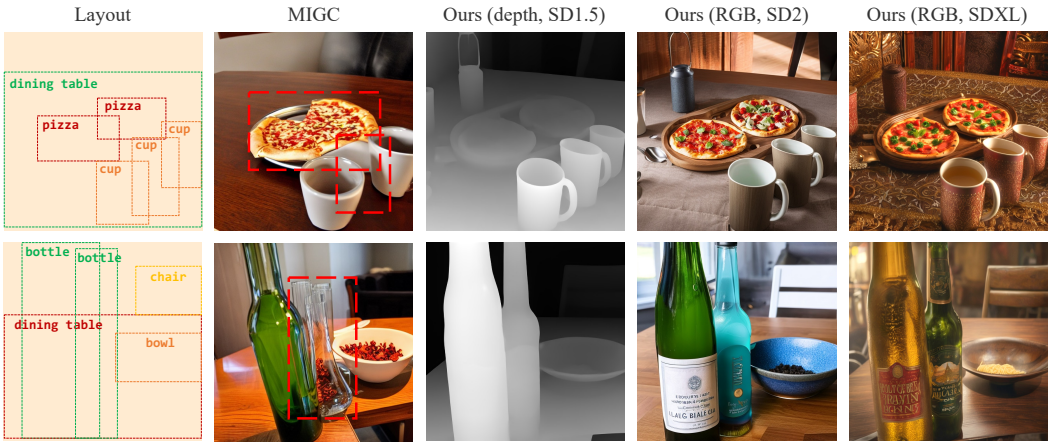


Figure 3: **Qualitative results on the COCO-Position (§4.3)**.

construction and the coarse-grained categories and structural properties of instances. We maintain the same batch size, learning rate, and other parameters as the previous work.

4.2 Experiment Setup

Baselines. We compared our proposed 3DIS method with state-of-the-art Multi-Instance Generation approaches. The methods involved in the comparison include training-free methods: BoxDiffusion (Xie et al., 2023) and MultiDiffusion (Bar-Tal et al., 2023); and adapter-based methods: GLIGEN (Li et al., 2023b), InstanceDiffusion (Wang et al., 2024), and MIGC (Zhou et al., 2024).

Evaluation Benchmarks. We conducted experiments using two widely adopted benchmarks, COCO-position (Lin et al., 2015) and COCO-MIG (Zhou et al., 2024), to assess the performance of models in different aspects of instance generation. The COCO-position benchmark emphasizes the evaluation of a model's capacity to control the spatial arrangement of instances, as well as their high-level categorical attributes. In contrast, the COCO-MIG benchmark is designed to test a model's ability to precisely render fine-grained attributes for each generated instance. To rigorously compare the models' performance in handling complex scene layouts, we concentrated our analysis on the COCO-position benchmark, specifically focusing on layouts containing five or more instances. For a comprehensive evaluation, each model generated 750 images across both benchmarks.

Evaluation Metrics. We used the following metrics to evaluate the model: *1) Mean Intersection over Union (MIoU)*, measuring the overlap between the generated instance positions and the target positions; *2) Local CLIP score*, assessing the visual consistency of the generated instances with their corresponding textual descriptions; *3) Average Precision (AP)*, evaluating the overlap between the generated image layout and the target layout; *4) Instance Attribute Success Ratio (IASR)*, calculating the proportion of correctly generated instance attributes; *5) Image Success Ratio (ISR)*, measuring the proportion of images in which all instances are correctly generated.

4.3 Comparison

Scene Construction. The results in Tab. 1 demonstrate the superior scene construction capabilities of the proposed 3DIS method compared to previous state-of-the-art approaches. Notably, 3DIS surpasses MIGC with an **11.8**% improvement in AP and a **16.3**% increase in AP_{75}, highlighting a closer alignment between the generated layouts and the user input. As shown by the visualizations in Fig. 3, 3DIS achieves marked improvements in scenarios with significant overlap, effectively addressing challenges such as object merging and loss in complex layouts. This results in the generation of a more accurate scene depth map, capturing the global scene structure with greater fidelity.

Detail Rendering. The results presented in Tab. 2 demonstrate that the proposed 3DIS method exhibits robust detail-rendering capabilities. Notably, the entire process of rendering instance attributes is **training-free** for 3DIS. Compared to the previous state-of-the-art (SOTA) training-free method, MultiDiffusion, 3DIS achieves a **30**% improvement in the Instance Attribute Success Ratio (IASR). Additionally, when compared with the SOTA adapter-based method, Instance Diffusion,

Table 2: **Quantitative results on proposed COCO-MIG-BOX (§4.3)**. $\mathcal{L}_i$ means that the count of instances needed to generate in the image is **i**.

Method	Instance Attribute Success Ratio↑						Mean Intersection over Union↑					
	$\mathcal{L}2$	$\mathcal{L}3$	$\mathcal{L}4$	$\mathcal{L}5$	$\mathcal{L}6$	$\mathcal{AVG}$	$\mathcal{L}2$	$\mathcal{L}3$	$\mathcal{L}4$	$\mathcal{L}5$	$\mathcal{L}6$	$\mathcal{AVG}$
Adapter rendering methods												
GLIGEN [CVPR23]	41.3	33.8	31.8	27.0	29.5	31.3	33.7	27.6	25.5	21.9	23.6	25.2
InstanceDiff [CVPR24]	61.0	52.8	52.4	45.2	48.7	50.5	53.8	45.8	44.9	37.7	40.6	43.0
MIGC [CVPR24]	74.8	66.2	67.4	65.3	66.1	67.1	63.0	54.7	55.3	52.4	53.2	54.7
***training-free** rendering*												
TFLCG [WACV24]	17.2	13.5	7.9	6.1	4.5	8.3	10.9	8.7	5.1	3.9	2.8	5.3
BoxDiff [ICCV23]	28.4	21.4	14.0	11.9	12.8	15.7	19.1	14.6	9.4	7.9	8.5	10.6
MultiDiff [ICML23]	30.6	25.3	24.5	18.3	19.8	22.3	21.9	18.1	17.3	12.9	13.9	15.8
3DIS (SD1.5)	65.9	56.1	55.3	45.3	47.6	53.0	56.8	48.4	49.4	40.2	41.7	44.7
3DIS (SD2.1)	66.1	57.5	55.1	51.7	52.9	54.7	57.1	48.6	46.8	42.9	43.4	45.7
3DIS (SDXL)	66.1	59.3	56.2	51.7	54.1	56.0	57.0	50.0	47.8	43.1	44.6	47.0
vs. MultiDiff	**+35**	**+34**	**+31**	**+33**	**+34**	**+33**	**+35**	**+31**	**+30**	**+30**	**+30**	**+31**
*rendering w/ **off-the-shelf** adapters*												
3DIS+GLIGEN	49.4	39.7	34.5	29.6	29.9	34.1	43.0	33.8	29.2	24.6	24.5	28.8
vs. GLIGEN	**+8.1**	**+5.9**	**+2.7**	**+2.6**	**+0.4**	**+2.8**	**+9.3**	**+6.2**	**+3.7**	**+2.7**	**+0.9**	**+3.6**
3DIS+MIGC	76.8	70.2	72.3	66.4	68.0	69.7	68.0	60.7	62.0	55.8	57.3	59.5
vs. MIGC	**+2.0**	**+4.0**	**+4.9**	**+1.1**	**+1.9**	**+2.6**	**+5.0**	**+6.0**	**+6.7**	**+3.4**	**+4.1**	**+4.8**

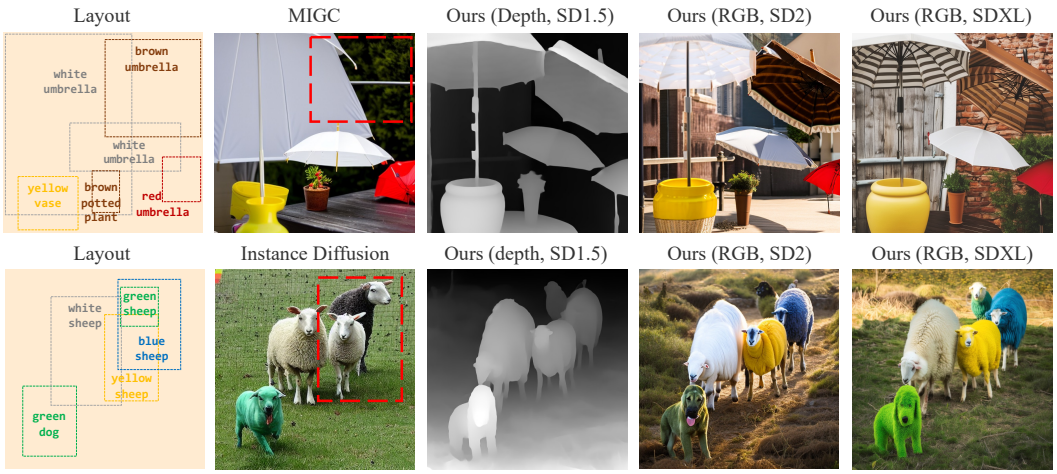


Figure 4: **Qualitative results on the COCO-MIG (§4.3)**.

which requires training for rendering, 3DIS shows a **5**% increase in IASR, while also allowing the use of higher-quality models, such as SD2 and SDXL, to generate more visually appealing results. Importantly, the proposed 3DIS approach is not mutually exclusive with existing adapter methods. For instance, combinations like 3DIS+GLIGEN and 3DIS+MIGC outperform the use of adapter methods alone, delivering superior performance. Fig. 4 offers a visual comparison between 3DIS and other SOTA methods, where it is evident that 3DIS not only excels in scene construction but also demonstrates strong capabilities in instance detail rendering. Furthermore, 3DIS is compatible with a variety of base models, offering broader applicability compared to previous methods.

4.4 Ablation Study

Constructing scenes with depth maps. Tab. 3 demonstrates that generating scenes in the form of depth maps, rather than directly producing RGB images, enables the model to focus more effectively on coarse-grained categories, structural attributes, and the overall scene composition. This approach leads to a **3.3**% improvement in AP and a **4.1**% increase in AP_{75}.

Tuning of the Text-to-depth model. Tab. 3 demonstrates that, compared to using LDM3D directly, fine-tuning LDM3D with pyramid diffusion as our base text-to-depth generation model

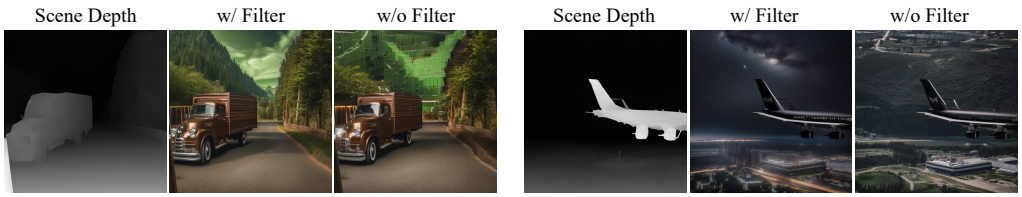


Figure 5: **Visualization of the Impact of Low-Pass Filtering on ControlNet (§4.4).**

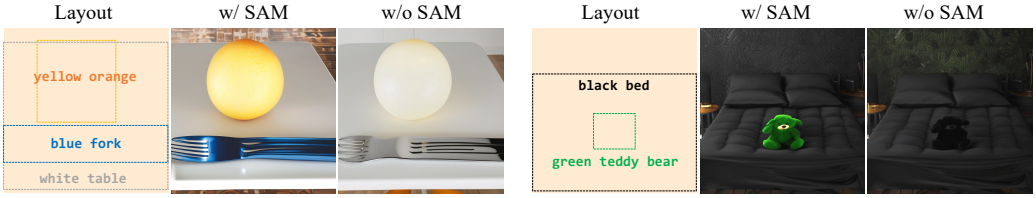


Figure 6: **Visualization of the Impact of SAM-Enhancing Instance Location (§4.4).**

results in a **1.3**% improvement in AP and a **2.2**% increase in AP_{75}. These improvements stem from the fine-tuning process, which encourages the depth generation model to focus more on recovering low-frequency components, benefiting the global scene construction.

Table 3: **Ablation study on scene generation (§4.4).**

method	AP/AP_{50}/AP_{75} ↑	MIoU ↑	FID ↓
w/o using depth	53.5 / 81.8 / 58.3	72.2	24.1
w/o aug data	54.0 / 78.4 / 59.4	73.3	23.5
w/o tuning LDM3D	55.5 / 81.9 / 60.2	72.8	25.2
w/ all	**56.8 / 82.3 /62.4**	**73.3**	**23.2**

Augmenting instance descriptions by removing adjectives. The data presented in Tab. 3 indicate that during the training of layout-to-depth adapters, augmenting instance descriptions by removing fine-grained attribute descriptions allows the model to focus more on the structural of the instances and the overall scene construction. This approach ultimately results in a **2.8**% improvement in AP and a **3.0**% increase in AP_{75}.

Table 4: **Ablation study on rendering (§4.4).**

method	IASR ↑	MIOU ↑	FID ↓
w/o Low-Pass Filter	55.87	46.93	24.50
w/o SAM-Enhancing	52.42	45.17	23.67
w/ all	**56.01**	**47.01**	**23.24**

Low-Pass Filtering on the ControlNet. Fig. 5 shows that filtering out high-frequency noise from ControlNet's feature maps improves the overall quality of the generated images, resulting in more accurate scene representation. Moreover, as indicated in Tab. 4, this process does not affect the Instance Attribute Success Ratio (IASR) and MIoU when rendering fine details.

SAM-Enhancing Instance Location. Fig. 6 illustrates that utilizing SAM for more precise instance location effectively prevents rendering conflicts caused by layout overlaps, ensuring accurate rendering of each instance's fine-grained attributes. As shown in Tab. 4, enhancing instance localization with SAM improves the Instance Attribute Success Ratio (IASR) by **3.19**% during rendering.

4.5 Universal rendering capabilities of 3DIS

Rendering based on different-architecture models. Fig. 1, 3, and 4 present the results of 3DIS rendering details using SD2 and SDXL without additional training. The results demonstrate that 3DIS not only leverages the enhanced rendering capabilities of these more advanced base models, compared to SD1.5, but also preserves the accuracy of fine-grained instance attributes.

Rendering based on different-style models. Fig. 7 presents the results of 3DIS rendering using various stylistic model variants (based on the SDXL architecture). As shown, 3DIS can incorporate scene depth maps to render images in diverse styles while preserving the overall structure and key instance integrity. Furthermore, across different styles, 3DIS consistently enables precise control over complex, fine-grained attributes, as illustrated by the third example in Fig. 7, where "Dotted colorful wildflowers, some are red, some are purple" are accurately represented.

Rendering Specific Concepts. 3DIS renders details leveraging pre-trained large models, such as SD2 and SDXL, which have been trained on extensive corpora. This capability allows users to render

Figure 7: **Rendering results based on different-style models (§4.5).**

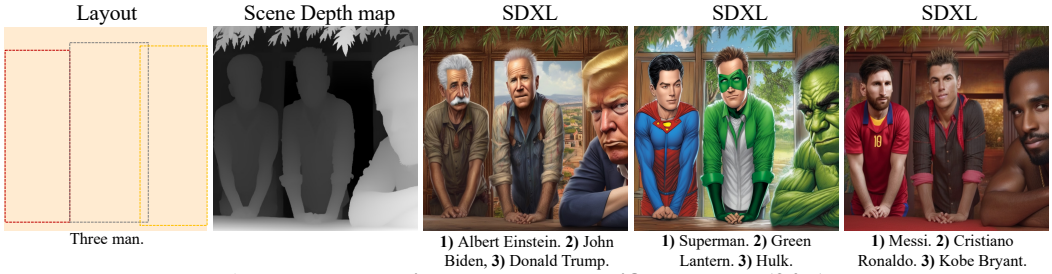


Figure 8: **Rendering results on specific concepts (§4.5).**

specific concepts. As demonstrated in Fig. 8, 3DIS precisely renders human details associated with specific concepts while preserving control over the overall scene.

5 Conclusion

We propose a novel 3DIS method that decouples image generation into two distinct phases: coarse-grained scene depth map generation and fine-grained detail rendering. In the scene depth map phase, 3DIS trains a Layout-to-Depth network that focuses solely on global scene construction and the coarse-grained attributes of instances, thus simplifying the training process. In the detail rendering phase, 3DIS leverages widely pre-trained ControlNet models to generate images based on the scene depth map, controlling the scene and ensuring that each instance is positioned accurately. Finally, our proposed detail renderer guarantees the correct rendering of each instance's details. Due to the training-free nature of the detail rendering phase, our 3DIS framework utilizes the generative priors of various foundational models for precise rendering. Experiments on the COCO-Position benchmark demonstrate that the scene depth maps generated by 3DIS create superior scenes, accurately placing each instance in its designated location. Additionally, results from the COCO-MIG benchmark show that 3DIS significantly outperforms previous training-free rendering methods and rivals state-of-the-art adapter-based approaches. We envision that 3DIS will enable users to apply a wider range of foundational models for multi-instance generation and be extended to more applications. In the future, we will continue to explore the integration of 3DIS with DIT-based foundational models.

REFERENCES

Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. *arXiv preprint arXiv:2302.08113*, 2023.

Jingye Chen, Yupan Huang, Tengchao Lv, Lei Cui, Qifeng Chen, and Furu Wei. Textdiffuser: Diffusion models as text painters, 2023. URL https://arxiv.org/abs/2305.10855.

Yutong Feng, Biao Gong, Di Chen, Yujun Shen, Yu Liu, and Jingren Zhou. Ranni: Taming text-to-image diffusion for accurate instruction following. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4744–4753, 2024.

Nocholas Guttenberg. Diffusion with offset noise, 2023. URL https://www.crosslabs.org/blog/diffusion-with-offset-noise.

Jonathan Ho. Classifier-free diffusion guidance. *ArXiv*, abs/2207.12598, 2022.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models, 2022. URL https://arxiv.org/abs/2206.00364.

Kasiopy. Multi-resolution noise for diffusion model training, 2023. URL https://wandb.ai/johnowhitaker/multires_noise/reports/. Last accessed 17 Nov 2023.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023.

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 *International conference on machine learning*, pp. 19730–19742. PMLR, 2023a.

Ming Li, Taojiannan Yang, Huafeng Kuang, Jie Wu, Zhaoning Wang, Xuefeng Xiao, and Chen Chen. Controlnet++: Improving conditional controls with efficient consistency feedback. *arXiv preprint arXiv:2404.07987*, 2024a.

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. *CVPR*, 2023b.

Zhen Li, Mingdeng Cao, Xintao Wang, Zhongang Qi, Ming-Ming Cheng, and Ying Shan. Photomaker: Customizing realistic human photos via stacked id embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8640–8650, 2024b.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

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.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2020.

Lingteng Qiu, Guanying Chen, Xiaodong Gu, Qi Zuo, Mutian Xu, Yushuang Wu, Weihao Yuan, Zilong Dong, Liefeng Bo, and Xiaoguang Han. Richdreamer: A generalizable normal-depth diffusion model for detail richness in text-to-3d. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9914–9925, 2024.

René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. *ICCV*, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. Stable diffusion version 2, 2023. URL https://stability.ai/news/stable-diffusion-v2-release.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *MICCAI*, abs/1505.04597, 2015.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

Gabriela Ben Melech Stan, Diana Wofk, Scottie Fox, Alex Redden, Will Saxton, Jean Yu, Estelle Aflalo, Shao-Yen Tseng, Fabio Nonato, Matthias Muller, et al. Ldm3d: Latent diffusion model for 3d. *arXiv preprint arXiv:2305.10853*, 2023.

Xudong Wang, Trevor Darrell, Sai Saketh Rambhatla, Rohit Girdhar, and Ishan Misra. Instancediffusion: Instance-level control for image generation, 2024.

Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. *arXiv preprint arXiv:2302.13848*, 2023.

Jiayu Xiao, Liang Li, Henglei Lv, Shuhui Wang, and Qingming Huang. R&b: Region and boundary aware zero-shot grounded text-to-image generation. *arXiv preprint arXiv:2310.08872*, 2023.

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. *ICCV*, 2023.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv preprint arXiv:2406.09414*, 2024.

Zhengyuan Yang, Jianfeng Wang, Zhe Gan, Linjie Li, Kevin Lin, Chenfei Wu, Nan Duan, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. Reco: Region-controlled text-to-image generation. In *CVPR*, 2023.

Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arxiv:2308.06721*, 2023.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, pp. 3836–3847, 2023.

Yuxuan Zhang, Jiaming Liu, Yiren Song, Rui Wang, Hao Tang, Jinpeng Yu, Huaxia Li, Xu Tang, Yao Hu, Han Pan, et al. Ssr-encoder: Encoding selective subject representation for subject-driven generation. *CVPR*, 2024.

Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. Unipc: A unified predictor-corrector framework for fast sampling of diffusion models. *NeurIPS*, 2023.

Dewei Zhou, You Li, Fan Ma, Zongxin Yang, and Yi Yang. Migc: Multi-instance generation controller for text-to-image synthesis. *CVPR*, 2024.

Appendix

A Inference Efficiency Analysis

Inference Efficiency Analysis of 3DIS. The 3DIS framework generates high-resolution images in three sequential stages: **1) The Layout-to-Depth Model**, which creates a coarse-grained scene depth map; **2) The Segmentation Model**, which extracts the precise shape of each instance from the scene depth map; **3) The Detail Renderer**, which uses various foundational models (SD2, SDXL, etc.) to produce the final high-resolution image. We evaluated the inference efficiency of these stages using an NVIDIA A100 GPU. Our test involved a layout with 10 instances, and we assessed the inference time for each stage over 50 runs to calculate an average time:

- **Layout-to-Depth Model:** Given that the global scene depth map does not require high granularity, the UniPCMultistepScheduler (Zhao et al., 2023) is employed for only 30 steps. The average time to generate a depth map is **5.66** seconds.
- **Segmentation Model:** We utilize the SAM model to segment the generated scene depth maps and get refined layouts. The refinement process by SAM takes **0.14** seconds.
- **Detail Renderer:** We use the EulerDiscreteScheduler (Karras et al., 2022) for 50 steps. The time for the SD1.5 model to render a 512×512 image is **5.27** seconds, the time for the SD2 model to render a 768×768 image is **11.28** seconds, and the time for the SDXL model to render a 1024×1024 image is **22.75** seconds.

Table A: **Average inference time of different layout-to-Image model.**

	GLIGEN	InstanceDiff	MIGC	3DIS (SD1.5)	3DIS (SD2)	3DIS (SDXL)
Inference Time (s)	12.75	42.48	6.81	11.07	17.08	28.55
Resolution	512	512	512	512	768	1024

Inference Efficiency Comparison. We conducted comparative experiments to evaluate the performance of various state-of-the-art (SOTA) methods, including GLIGEN (Li et al., 2023b), Instance Diffusion (Wang et al., 2024), and MIGC (Zhou et al., 2024), using NVIDIA A100 GPU. All models were tested using the default configurations in their GitHub repositories. We evaluated the inference efficiency of these stages using an NVIDIA A100 GPU. Our test involved a layout with 10 instances, and we assessed the inference time for each stage over 50 runs to calculate an average time. The experimental results are shown in Tab. A. The conclusions are as follows:

- **3DIS demonstrates faster inference speeds with SD1.5.** Since the scene depth map generated by 3DIS does not require too high granularity, the speed of generating the scene depth map is very fast. The average inference time of 3DIS + SD1.5 is 11.07s, even faster than GLIGEN and Instance Diffusion, which are based on the same SD1.5 base model.
- **3DIS demonstrates acceptable inference speeds with SD2 and SDXL.** As we increase model capacity and image resolution, the inference time for 3DIS also rises. Rendering times are **17.08** seconds for SD2 and **28.55** seconds for SDXL, which we consider to be acceptable. Additionally, our experiments show that using 3DIS with SDXL even achieves faster processing speeds than InstanceDiffusion. As discussed in Section 4.3, the performance of 3DIS + SDXL on COCO-MIG slightly surpasses that of InstanceDiffusion, demonstrating the practicality and efficiency of our 3DIS framework comprehensively.

B Results of Overlapping Layouts with Depth Ambiguity

3DIS allows for direct adjustment of the instance font-back according to user specifications (see Fig. A). Although our layout-to-depth model does not explicitly incorporate instance front-back ordering during the training process or network design, we found that certain training-free methods can still achieve control over instance front-back ordering. Specifically, our layout-to-depth model integrates layout information via a layout adapter (i.e., MIGC). For N instances, this adapter

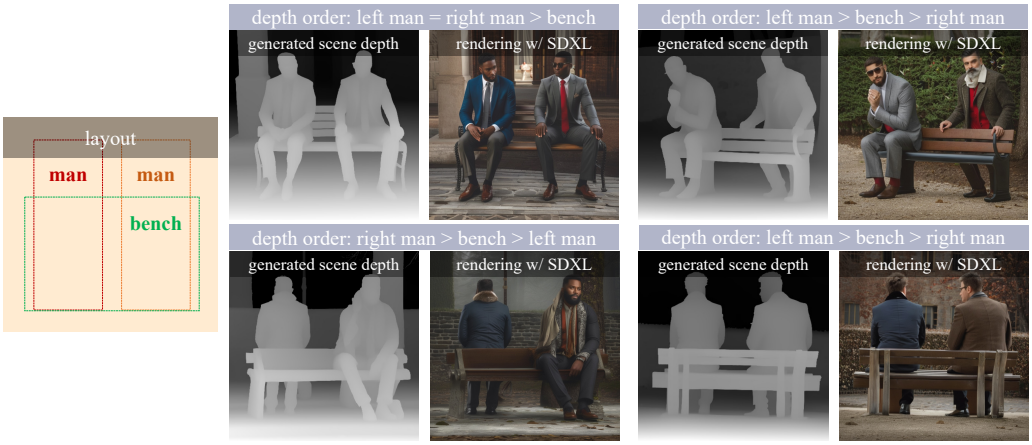


Figure A: **User-specified Front-Back Instance Ordering in Scene Depth Map Generation (§B).** For layouts with depth ambiguity, 3DIS allows for direct adjustment of the instance ordering according to user specifications, generating distinct scene depth maps and rendering them accordingly.

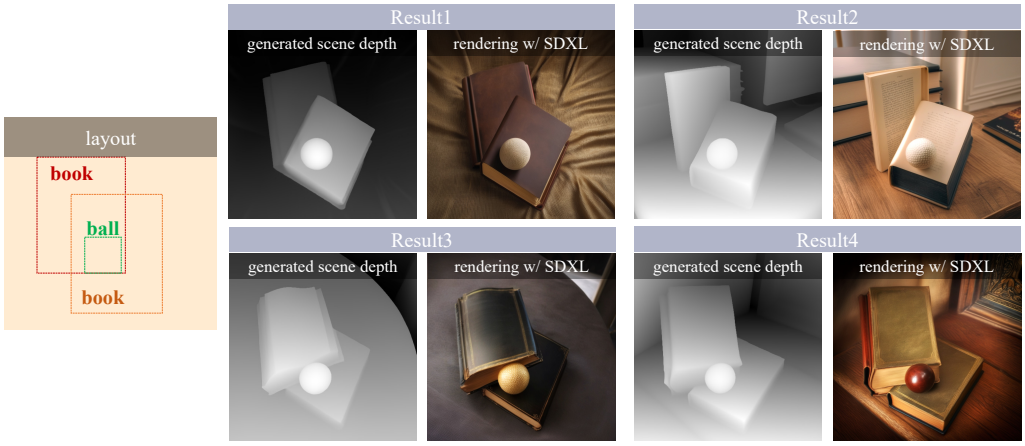


Figure B: **Automatic Front-Back Instance Ordering in Scene Depth Map Generation. (§B).** For the same overlapping layout with depth ambiguity, 3DIS can generate different scene depth maps with varying seeds, ensuring that the generated scenes adhere to the specified layout. Instances overlapping in the layout may display varying front-back order across different generated outcomes.

encodes them into N tokens, which are then injected into image features through a newly trainable Cross-Attention layer. For each specific pixel in the image features, the Cross-Attention layer uses a softmax function to determine the scale score of each instance token. Notably, we discovered that by adjusting the scale score (before the softmax function) of a token, we can control the relative depth ordering of instances (e.g., larger scale scores bring instances to the foreground, while smaller scale scores push them to the background). By adjusting the scale scores for each instance, we can thus control the front-back ordering within overlapping regions of the scene.

3DIS is capable of automatically adjusting the depth order of instances without explicit specifications (see Fig. B). As illustrated in Fig. B, the overlap of instances can be categorized into two types: 1) Complete overlap, as seen in the relationship between the ball and the books. As the ball's bounding box is fully enclosed within the books' bounding boxes, 3DIS typically generates it in the foreground to prevent it from disappearing. 2) Partial overlap, as in the case of the two books. In this scenario, depending on the seed, the front-back ordering of the books may vary, resulting in different depth placements across the generated scenes.

C Comparison of LDM3D and RichDreamer

Upon investigation, we identified RichDreamer (Qiu et al., 2024) and LDM3D (Stan et al., 2023) as the primary models employed for text-to-depth generation. To compare their performance, we

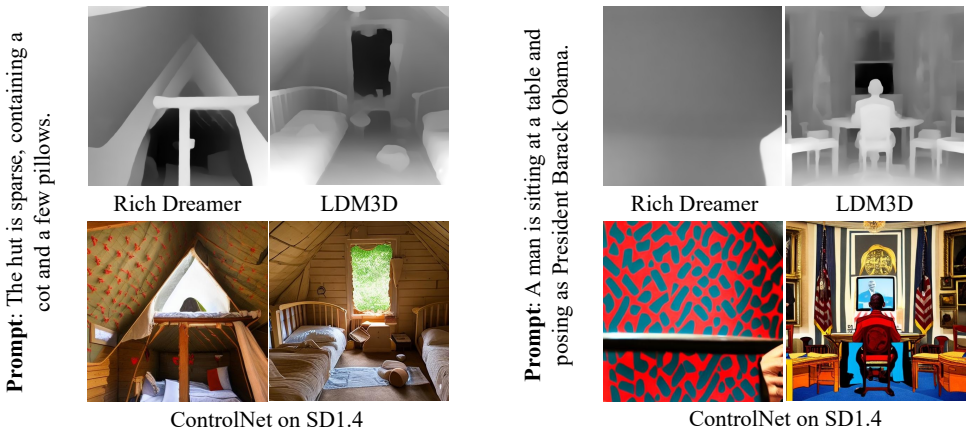


Figure C: **Comparison of LDM3D and RichDreamer.**

utilized prompts from the COCO2014 dataset as input for both models, with the corresponding results illustrated in Fig. C. Our analysis indicates that LDM3D demonstrates a superior ability to preserve the original SD1.4 priors, resulting in enhanced text comprehension and more precise control over scene generation. In contrast, RichDreamer exhibits certain shortcomings: (i) it often misses semantic details or omits entire objects in the depth maps, as seen in cases where essential elements like the **cot** and **man** are entirely absent; (ii) the depth maps produced by RichDreamer frequently suffer from artifacts such as blotches or thread-like distortions, particularly when used in conjunction with ControlNet. Therefore, after a thorough comparison, we selected LDM3D as the base model for text-to-depth generation in our 3DIS system.

D Visualization on the impact of the LDM3D Fine-Tuning

Although LDM3D is capable of generating relatively good depth maps, several issues remain: (i) Since LDM3D was trained using depth maps extracted from the DPT-Large Model (Ranftl et al., 2021) , the resulting image quality is relatively poor. (ii) As a diffusion model trained by Gaussian noise, LDM3D exhibits limited ability to recover low-frequency content (Guttenberg, 2023). This is clearly illustrated in Fig. D, where the generated depth maps struggle to produce large uniform color blocks. Moreover, the average color value of the depth maps tends to converge towards the initial noise, whose mean value is close to 0. This constraint places a harmful limitation on text-to-depth generation.

To address (i), we fine-tuned the model using depth maps extracted from the latest Depth-Anything V2 model. For (ii), we adopted pyramid noise instead of Gaussian noise, which helps mitigate the constraints on text-to-depth generation. As shown in Fig. D, the fine-tuned LDM3D model is capable of generating depth maps with higher contrast and improved overall quality.

E Examples of generated annotation

Text-depth pair in LAION-art (see Fig. E). The text-to-depth pair is essential for training our text-to-depth model. To obtain high-quality RGB images, we selected images from LAION-art with an aesthetic score greater than 8.0 and a resolution exceeding 512. Given that the text descriptions in LAION-art are often noisy, we chose to use the BLIP2 (Li et al., 2023a) model to generate more accurate captions. As shown in Fig. E, BLIP-generated captions can precisely capture the key information of the image. While the model still has limitations in describing certain fine-grained attributes—such as the color in the first example of the second row, where the description is inaccurate—this is not crucial for depth map generation, where fine-grained details are less significant. We use the Depth Anything V2 model to obtain high-quality depth maps corresponding to each image, which, together with the generated captions, form the text-depth pairs for training.

Layouts in COCO dataset (see Fig. F). The COCO (Lin et al., 2015) dataset contains images along with corresponding human-annotated natural language descriptions. For example, in the first image

of the first row of Fig. F, the annotated description is: "A white vase filled with a mix of white and pink flowers on a porch railing." To further extract descriptions for each instance, we use the Stanza (Qi et al., 2020) parser to analyze the noun phrases in the sentence, such as "A white vase," "A mix of white and pink flowers," and "porch railing." Based on these instance descriptions, we employ Grounding-DINO (Liu et al., 2023) to detect the bounding boxes of each instance, thereby obtaining the layout of the entire image and detailed descriptions of the instances.

F User Study

We conducted a user study to evaluate user preferences, selecting three methods for comparison: 3DIS, MIGC (Zhou et al., 2024), and InstanceDiffusion (Wang et al., 2024). For each participant in the user study, we randomly selected 30 images from the COCO-MIG benchmark and asked them to rank the images based on their preference. A total of 30 participants were invited, and the aggregated results are presented in Fig. G. The results indicate that, compared to MIGC and InstanceDiffusion, 3DIS was generally preferred by users. This preference is attributed partly to 3DIS's superior control over spatial positioning and also to its ability to leverage stronger foundational models for rendering in a training-free manner, resulting in higher-quality images.

G Additional Examples of 3DIS

Additional examples of controlling shape and pose (see Fig. H). Under the same layout, 3DIS can generate different scene depth maps and control coarse-grained attributes of different instances, such as shape and pose. As shown in Fig. H(a), we can freely change the shape of the cake and table within the same layout. Similarly, in Fig. H(b), we can adjust each person's pose.

Additional examples of complicated layouts (see Fig. I). For highly complex layouts, 3DIS reliably ensures accurate generation results. In Fig. I(a), 3DIS successfully creates a counterfactual scene where an ice mountain, volcano, mallard, swallow, and cherry coexist harmoniously. In Fig. I(b), 3DIS precisely renders each part of an eagle according to the specified input.

Additional examples of COCO-position benchmark. Fig. J presents additional results of scene depth map generation using our 3DIS system. The results demonstrate that, even with complex layouts, 3DIS effectively understands and generates cohesive scenes, harmoniously placing all objects within them. Furthermore, even in cases of significant overlap, such as the five suitcases in the fifth row, 3DIS handles the arrangement with precision, maintaining clear object separation and preventing blending.

Additional examples of COCO-MIG benchmark. Fig. L presents additional results of 3DIS on the COCO-MIG dataset, revealing several key advantages over the previous state-of-the-art model, MIGC. *1) 3DIS demonstrates superior scene construction capabilities*, as seen in the first and second rows, where it constructs more coherent scenes that appropriately place all specified instances—such

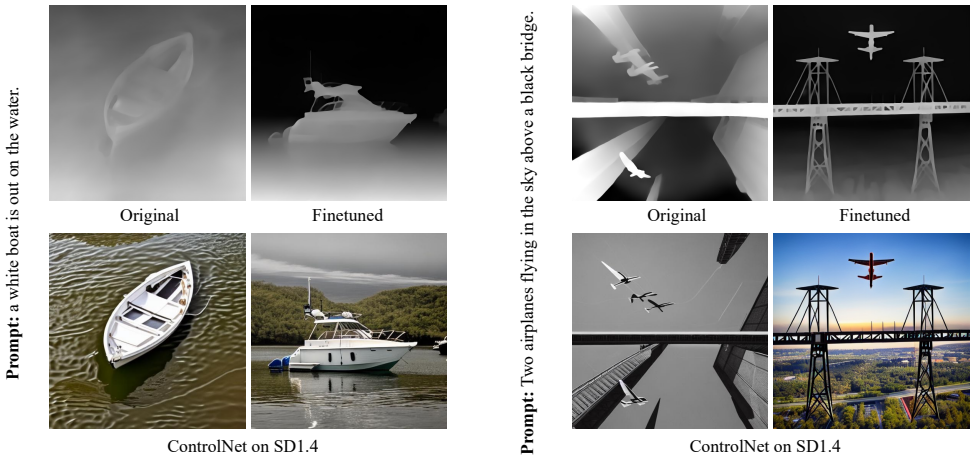


Figure D: **Comparison of original LDM3D and finetuned LDM3D.**

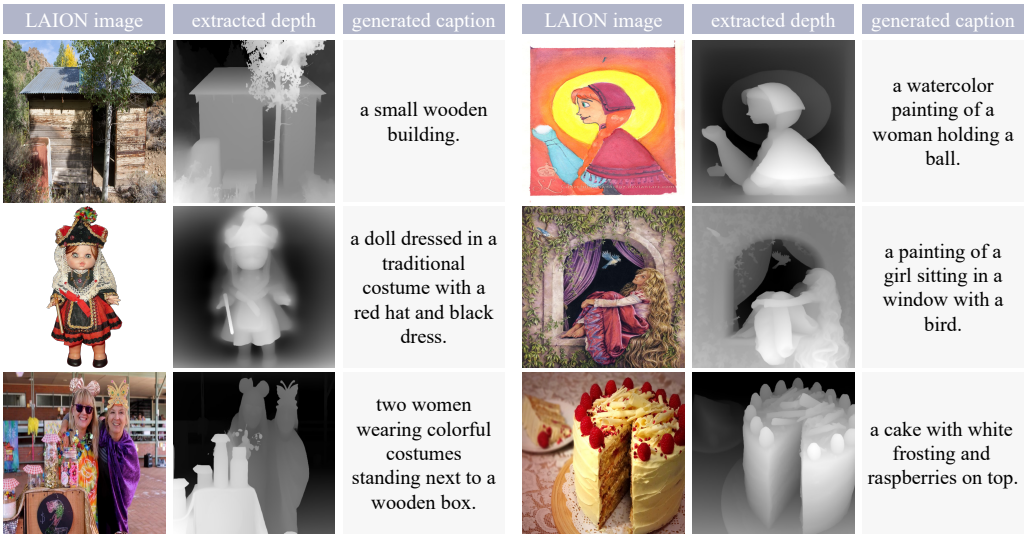


Figure E: **Examples of the generated annotation in the LAION-art dataset.** By utilizing the Depth Anything V2 model to extract depth maps and employing the BLIP2 model to generate captions corresponding to images, we can obtain high-quality text-depth pairs. These pairs will be used to train our text-to-depth model.

Figure F: **Examples of the generated layouts in the COCO dataset.** We have omitted the adjectives from each instance to better highlight the generated layout.

as rendering an indoor environment when prompted with "refrigerator." *2) 3DIS exhibits enhanced detail rendering*, as shown in the fourth to sixth rows. By leveraging the more advanced SDXL model in a training-free manner, 3DIS outperforms MIGC, which primarily relies on SD1.5, producing more visually appealing and structurally refined results. *3) 3DIS handles smaller instances better*, as demonstrated in the third row with the "red bird" and "yellow dog." Its ability to render at higher resolutions using SDXL leads to clearer and more accurate depictions of these smaller objects. **Finally, 3DIS excels in managing overlapping objects**, as illustrated in the seventh row, where it avoids object merging while generating the scene's depth map.

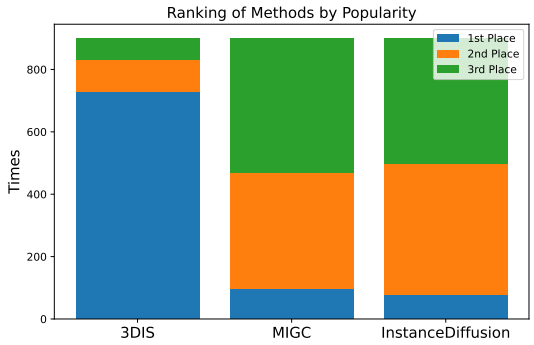
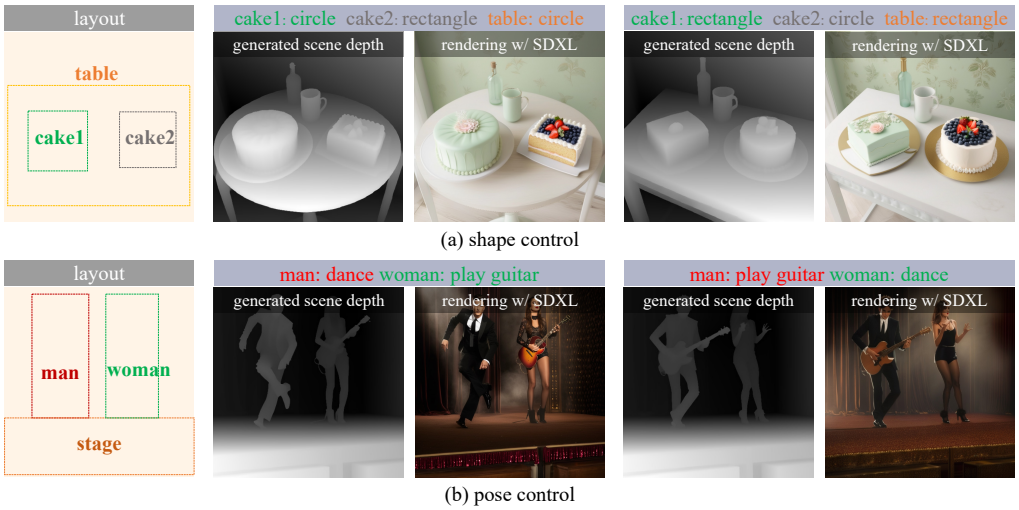


Figure G: **User Study.** Compared with the previous state-of-the-art methods, 3DIS is more popular.



(a) shape control

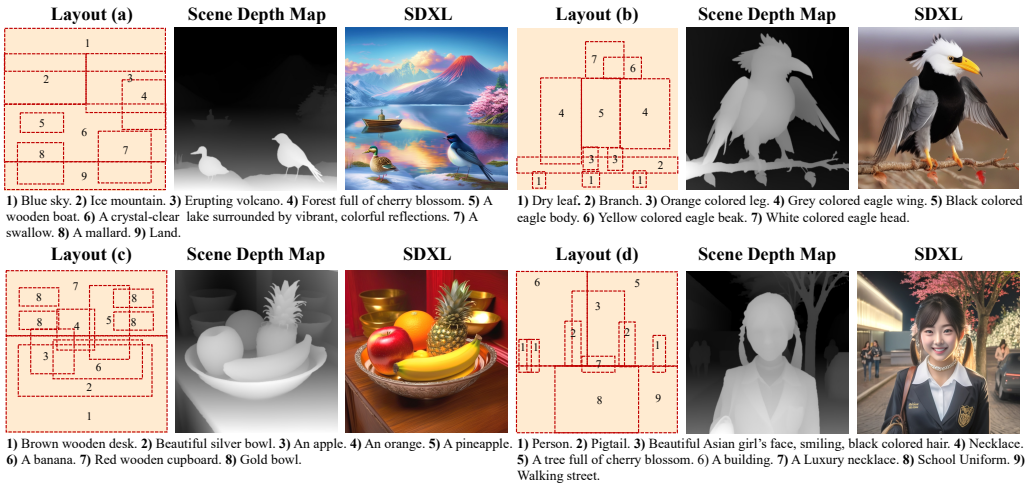
(b) pose control

Figure H: **Additional Generated Examples.** With the same layout, 3DIS can modify the shape and pose of each instance automatically.

H More Details of the Inference Pipeline

Scene Depth Maps Generation. Given that the scene depth map primarily focuses on coarse-grained attributes for scene construction and instance placement, it is unnecessary to generate extensive detail at this stage. Therefore, unlike previous methods (Zhou et al., 2024; Li et al., 2023b), which typically employ 50 steps for scene generation, we use only 30 steps, utilizing the UniPCMultistepScheduler (Zhao et al., 2023). Additionally, the Classifier-Free Guidance (Ho, 2022) (CFG) scale is set to 7.5.

Detail Rendering. In this phase, we utilize the EulerDiscreteScheduler (Karras et al., 2022) for 50 steps to render details meticulously. To reduce high-frequency noise in the generated depth map and to emphasize low-frequency scene information, we apply an FFT filter to the ControlNet signals. This filtering is specifically targeted at the mid and lower resolution upper layers. Initially, we perform a Fast Fourier Transform (FFT) to centralize the zero-frequency component within the spectrum. Subsequently, we design and implement a frequency mask that attenuates high frequencies beyond the central region extending to H/4 and W/4 from the center, setting a scale of 0.5 to predominantly preserve the central region, where H and W represent the height and width of the residual features injected from the ControlNet. An inverse FFT is then conducted to transform the data back to the spatial domain. The outcome is a refined version of the ControlNet feature, enriched with primarily low-frequency scene information.



1) Blue sky. **2)** Icc mountain. **3)** Erupting volcano. **4)** Forest full of cherry blossom. **5)** A wooden boat. **6)** A crystal-clear lake surrounded by vibrant, colorful reflections. **7)** A swallow. **8)** A mallard. **9)** Land.

1) Dry leaf. **2)** Branch. **3)** Orange colored leg. **4)** Grey colored eagle wing. **5)** Black colored eagle body. **6)** Yellow colored eagle beak. **7)** White colored eagle head.

1) Brown wooden desk. **2)** Beautiful silver bowl. **3)** An apple. **4)** An orange. **5)** A pineapple. **6)** A banana. **7)** Red wooden cupboard. **8)** Gold bowl.

1) Person. **2)** Pigtail. **3)** Beautiful Asian girl's face, smiling, black colored hair. **4)** Necklace. **5)** A tree full of cherry blossom. 6) A building. **7)** A Luxury necklace. **8)** School Uniform. **9)** Walking street.

Figure I: **Additional Generated Examples.** 3DIS also demonstrates robust generation capabilities for complex layouts.

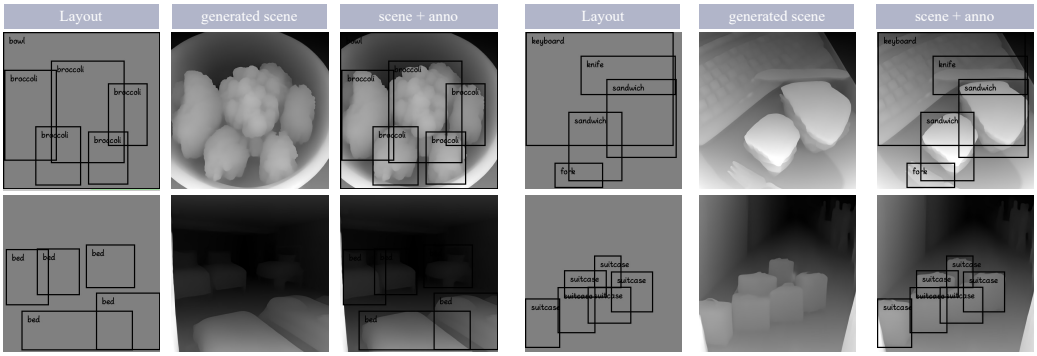
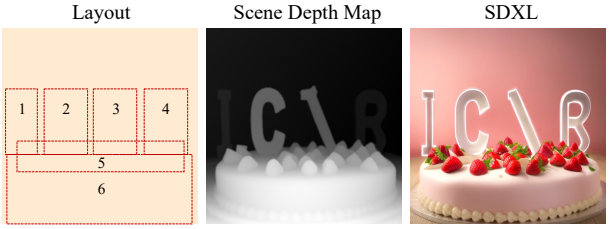


Figure J: **More results of the generated scene depth map.**

I LIMITATION

Although 3DIS leverages various foundation models for rendering fine instance details, its scene construction continues to rely on the less advanced SD1.5 model. This dependency limits 3DIS's capacity to accurately generate complex structures, particularly in tasks that SD1.5 struggles with, such as text rendering, intricate shapes, or highly detailed spatial configurations. For example, if we aim to generate a high-quality strawberry cake with the text "ICLR" written on it, 3DIS is unlikely to generate scene depth maps correctly (e.g., the wrong "L" letter in Fig. K). Addressing this limitation in future work could involve the development of specialized datasets aimed at enhancing the model's proficiency in handling complex structures, such as MARIO-10M (Chen et al., 2023), thereby improving the overall robustness and versatility of 3DIS in a broader range of applications.



1) Letter "I". 2) Letter "C". 3) Letter "L". 4) Letter "R". 5) A huge delicious cake. 6) Many red strawberries.

Figure K: **Failure case of the 3DIS.**

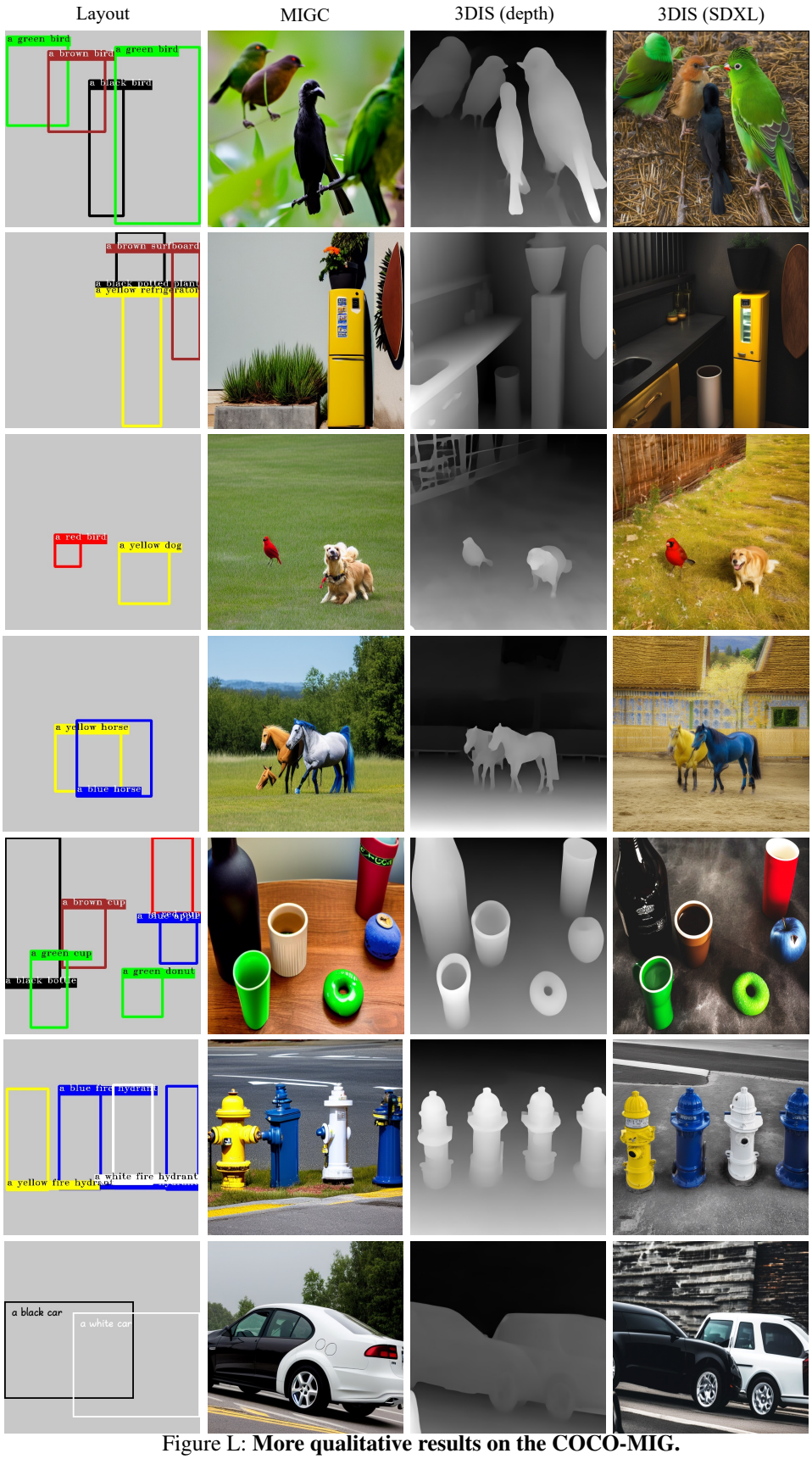


Figure L: **More qualitative results on the COCO-MIG.**