# JointDiT: Enhancing RGB-Depth Joint Modeling with Diffusion Transformers

Kwon Byung-Ki[1,2†] Qi Dai[2] Lee Hyoseok[1] Chong Luo[2] Tae-Hyun Oh[3]

[1]POSTECH [2]Microsoft Research Asia [3]KAIST

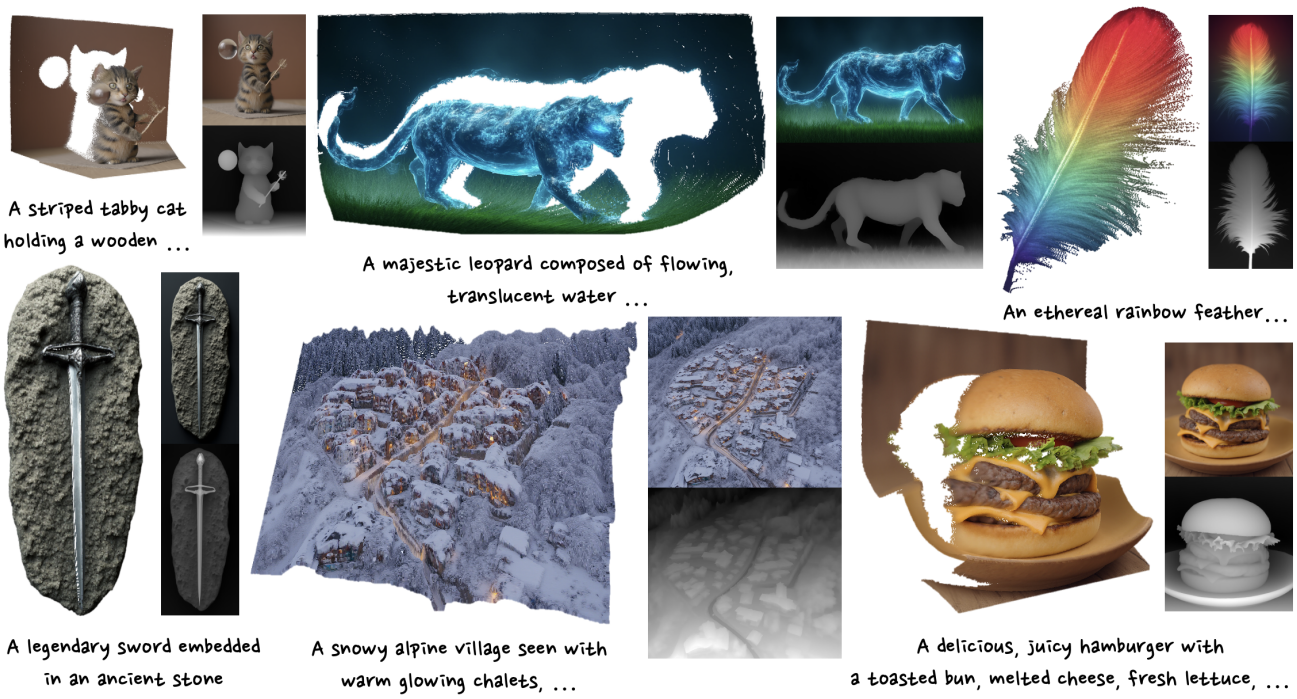


Figure 1. We present JointDiT, a diffusion transformer modeling the RGB-Depth joint distribution. By leveraging the image prior of a state-of-the-art diffusion transformer [7], JointDiT generates high-fidelity images and geometrically plausible and accurate depth maps.

## Abstract

*We present JointDiT, a diffusion transformer that models the joint distribution of RGB and depth. By leveraging the architectural benefit and outstanding image prior of the state-of-the-art diffusion transformer, JointDiT not only generates high-fidelity images but also produces geometrically plausible and accurate depth maps. This solid joint distribution modeling is achieved through two simple yet effective techniques that we propose, namely, adaptive scheduling weights, which depend on the noise levels of each modality, and the unbalanced timestep sampling strategy. With these techniques, we train our model across all noise levels for each modality, enabling JointDiT to naturally handle various combinatorial generation tasks, including joint generation, depth estimation, and depth-conditioned image generation by simply controlling the timesteps of each branch. JointDiT demonstrates outstanding joint generation performance. Furthermore, it achieves comparable results in depth estimation and depth-conditioned image generation, suggesting that joint distribution modeling can serve as a viable alternative to conditional generation. The project page is available at* https://byungki-k.github.io/JointDiT/

## 1. Introduction

In the era of generative AI, diffusion models have made remarkable advancements in synthesizing images [26, 52]. The outstanding capability of text-to-image diffusion models has been found to be useful not only for image generation but also for solving important inverse problems [12–14, 28], image inpainting [15, 44], image editing [16, 31],

[†]Work done during an internship at Microsoft Research Asia.

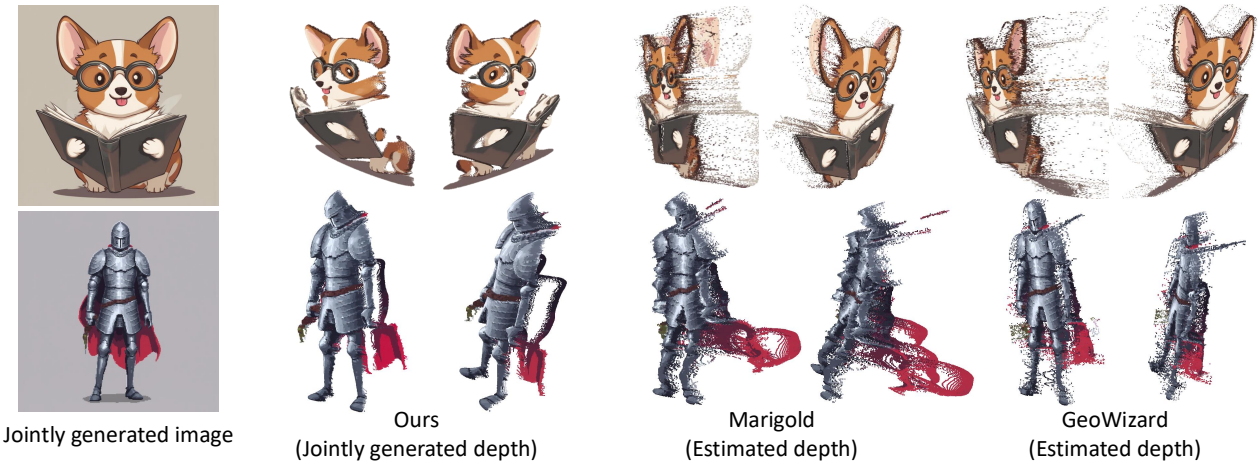


Figure 2. **3D lifting results of JointDiT and depth diffusion models, *i.e.*, Marigold [32] and GeoWizard [21].** JointDiT also shows more plausible 3D point clouds than depth estimation models in challenging illustration domains, likely due to the complementary behavior of the RGB and depth branches across generative processes, *i.e.*, the RGB branch focuses on texture, and the depth branch on structure.

and even further cross-modal conditional generation, such as depth-conditioned image generation [6, 45, 72], and image-conditioned depth estimation [21, 22, 32]. These works have shown that using the image prior of diffusion models is effective for modeling conditional distribution.

Recently, in image and depth modalities, joint distribution modeling [35, 60, 71] has shown that it not only enables joint generation but also shows potential as a viable alternative to existing depth estimation methods and depth-conditional image generation methods within a single unified framework by treating them as special cases of joint distribution modeling. It demonstrates that joint modeling can be easily generalized for various tasks including controllable and conditional generation and estimation. Despite this versatility, the realism of the generation is limited.

In this work, we propose JointDiT, a diffusion transformer designed for solid joint distribution modeling of image and depth. Figure 1 demonstrates the high-fidelity joint generation results of JointDiT. The high-fidelity images and geometrically accurate depth maps visually highlight the joint distribution capability of JointDiT, which has not been achieved. Furthermore, we design JointDiT to provide a replaceable alternative to conditional distribution models by constructing a joint distribution at all noise levels for each modality. For instance, the model performs joint generation when both the image and depth map are noise, depth estimation when only the image is clean, and depth-conditioned image generation when only the depth map is clean.

To achieve this, we model the joint distribution by harnessing the strong image prior of a state-of-the-art diffusion transformer [7] and building a parallel depth branch through joint connection modules. By training on separate noise levels for each modality, JointDiT flexibly facilitates combinatorial tasks of image and depth by simply controlling the timestep of each branch. To enable separate noise level training, we propose two simple yet effective techniques, *i.e.*, *adaptive scheduling weights* and *unbalanced timestep sampling strategy*, designed for multi-modal diffusion training with separate noise levels. JointDiT achieves significantly superior joint generation results compared to previous joint generation methods [35, 60, 71] while demonstrating comparable performance in conditional generation tasks, such as depth estimation and depth-conditioned image generation. JointDiT also enables plausible depth generation even for challenging domains such as cartoon images and pixel art illustrations, where depth estimation methods [21, 32] often struggle, as shown in Fig. 2. We further observe that the RGB and depth branches adopt complementary behavior in the joint generation process, with the depth branch capturing structural information and the RGB branch focusing on complementary aspects related to texture and appearance, which may underlie the plausible results observed in challenging domains.

We summarize our contributions as follows:

- We present JointDiT, a model for solid joint distribution modeling between image and depth modalities across all noise levels by leveraging the image prior of diffusion transformers. It supports combinatorial tasks, such as joint generation, depth estimation, and depth-conditioned image generation via simple timestep control.
- We propose adaptive scheduling weights and unbalanced timestep sampling strategy for separate noise level training in multi-modality, which significantly improves performance on combinatorial tasks. Through these techniques, we demonstrate that joint distribution modeling

is a viable alternative to conditional generation.

## 2. Preliminaries

**Flow matching.** Flow matching is generative modeling that learns a time-dependent vector field, which transports one probability distribution to another. It is closely related to Continuous Normalizing Flows (CNFs) [11], which model such transformations via differential equations. We adopt the notation from Lipman *et al*. [39] to describe the flow matching formulation and objective. Given the data points $x \in \mathbb{R}^d$ and probability density path $p : [0, 1] \times \mathbb{R}^d \rightarrow \mathbb{R}_{>0}$ that is a time-dependent probability density function satisfying $\int p_t(x)dx = 1$, the time-dependent vector field $v : [0, 1] \times \mathbb{R}^d \rightarrow \mathbb{R}^d$ combines with a flow $\phi : [0, 1] \times \mathbb{R}^d \rightarrow \mathbb{R}^d$, leading to the ordinary differential equation (ODE):

$$\frac{d}{dt}\phi_t(x) = v_t(\phi_t(x)), \quad \phi_0(x) = x, \tag{1}$$

where $\phi_0(x) = x$ is an initial condition. Through the push forward equation, the probability density function at time $t$, *i.e*., $p_t$, is transformed by $p_t = [\phi_t]_* p_0$. The $*$ and $p_0$ represent the push forward operator and simple prior, *e.g*., the standard normal distribution, and $p_1$ represents a data distribution. The objective of flow matching is to estimate $v_t(x)$ using a learnable neural network $v_{t,\theta}(x)$ by minimizing:

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{t,\, p_t(x)}\big[\|v_{t,\theta}(x) - v_t(x)\|\big]. \tag{2}$$

However, obtaining the true vector field $v_t$ is intractable. To address this, Lipman *et al*. [39] proposed Conditional Flow Matching (CFM), which introduces a condition by sampling the accessible data sample $x_1$ from the unknown data distribution $q(x_1)$. By conditioning the true vector field on $x_1$, that is $v_t(x|x_1)$, a tractable objective is obtained, as follows:

$$\mathcal{L}_{\text{CFM}} = \mathbb{E}_{t,\, q(x_1), p_t(x|x_1)}\big[\|v_{t,\theta}(x) - v_t(x|x_1)\|\big]. \tag{3}$$

## 3. Method

Our goal is to develop a unified network that models the joint distributions between images and depth maps across all noise levels. This network can be applied to various tasks, including joint image-depth generation, depth estimation from an image, and depth-conditioned image generation by adjusting the noise levels of each modality.

To achieve this, inspired by previous works [8, 35] that employ separate noise sampling for images and conditions, we extend the flow matching framework to learn a joint vector field $v_{t_x,t_y}(x, y|x_1, y_1)$ with two independent timesteps, $t_x$ and $t_y$. Here, $x$ and $y$ represent data points sampled from the RGB image and depth map distributions, respectively. To estimate $v_{t_x,t_y}(x, y|x_1, y_1)$, we design a learnable neural network $v_{t_x,t_y,\theta}(x, y)$ and train it by minimizing the following Joint Conditional Flow Matching (JCFM) loss:

$$\mathcal{L}_{\text{JCFM}}(\theta) = \mathbb{E}_{t_x,t_y,q(x_1,y_1),p_{t_x,t_y}(x,y|x_1,y_1)}\Big[\| v_{t_x,t_y,\theta}(x, y) - v_{t_x,t_y}(x, y|x_1, y_1)\|\Big]. \tag{4}$$

Once the network successfully learns to estimate the vector field $v_{t_x,t_y}(x, y|x_1, y_1)$, various tasks can be performed simply by adjusting $t_x$ and $t_y$ without any additional guidance. For example, initially setting $t_x = 0, t_y = 0$ leads to the joint generation of both images and depth maps. When $t_x = 1, t_y = 0$, it performs depth estimation from a given image, and when $t_x = 0, t_y = 1$, it becomes a depth-conditioned image generation. In the later section, we will denote the noisy image and depth map samples $(x, y) \sim p_t(x, y)$ as $x_t$ and $y_t$ for simplicity.

### 3.1. Joint Diffusion Transformer (JointDiT)

Figure 3 shows JointDiT architecture. JointDiT is built on Flux [7], an advanced diffusion transformer model that consists of multi-modal diffusion transformer (MM-DiT) and parallel diffusion transformer (P-DiT) blocks [18, 19]. To harness its strong image prior and the benefits of transformer architectures in dense prediction tasks [50], we extend it to joint image and depth distribution modeling by introducing a parallel depth branch alongside the pre-trained RGB branch. Thereafter, we add LoRAs [27] to the MM-DiT and P-DiT blocks to process the additional depth domain. Additionally, joint connection modules are introduced in each DiT block to model joint distribution by interchanging features between the RGB and depth branches. We train the LoRAs and joint connection modules while keeping the pre-trained backbone model frozen.

In the joint connection modules (see Fig. 3-b), feature exchange for joint distribution modeling occurs within the attention mechanism of each DiT block. We adopt the joint cross-attention module from the prior work [35]. This module facilitates joint distribution training by exchanging queries between the RGB and depth branches through attention mechanisms. The motivation is that self-attention plays a key role in the form and structure of images [63].

To further reinforce this motivation, we propose adaptive scheduling weights, which encourage the joint model to follow the form and structure of the relatively cleaner domain between the RGB and depth branches. Specifically, we adaptively schedule the amount of information transferred between branches by joint cross attention according to the relative cleanliness of the given noisy image $x_{t_x}$ and the noisy depth $y_{t_y}$. This approach is also intuitively reasonable, as cleaner data inherently provides more useful information for joint generation. The adaptive schedul-

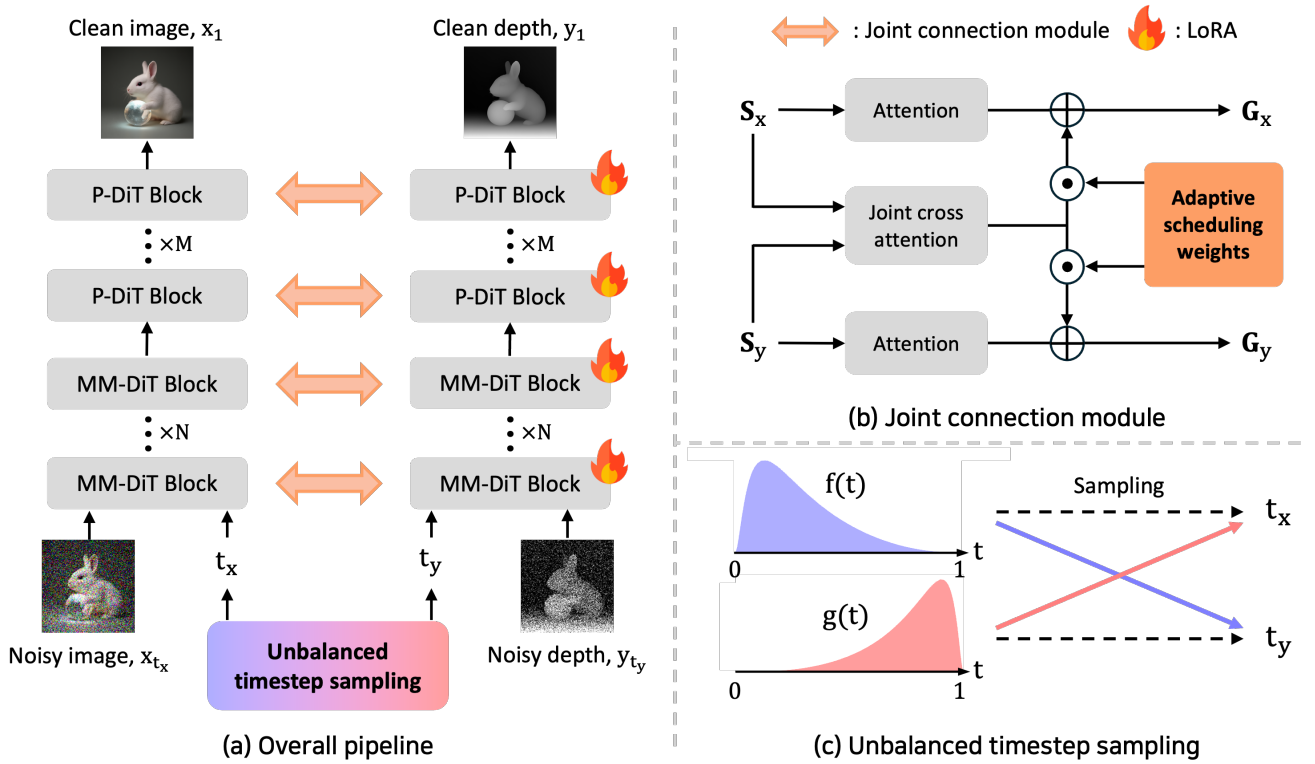


Figure 3. **Overall pipeline of JointDiT.** Building on Flux [7], we introduce a parallel depth branch with trainable LoRAs [27]. The joint connection module enables the aligned joint generation. We propose adaptive scheduling weights and an unbalanced timestep sampling strategy for effective training using separate timesteps. MM-DiT and P-DiT denote the multi-modal [19] and parallel [18] diffusion transformers, respectively. The two timestep distributions, $f(t)$ and $g(t)$, are described in the supplementary material.

ing weights are individually multiplied with the joint cross-attention outputs, which corresponds to:

$$\begin{aligned} \mathbf{G}_x &= \text{Attn}(\mathbf{S}_x) + w_x(t_x, t_y) \cdot \text{JointAttn}(\mathbf{S}_x, \mathbf{S}_y), \\ \mathbf{G}_y &= \text{Attn}(\mathbf{S}_y) + w_y(t_x, t_y) \cdot \text{JointAttn}(\mathbf{S}_x, \mathbf{S}_y). \end{aligned} \tag{5}$$

$w_x$ and $w_y$ are adaptive scheduling weights for:

$$\begin{aligned} w_x(t_x, t_y) &= \text{sigmoid}\left(\alpha\left(\frac{t_y}{t_x + t_y} - \frac{1}{2}\right)\right), \\ w_y(t_x, t_y) &= \text{sigmoid}\left(\alpha\left(\frac{t_x}{t_x + t_y} - \frac{1}{2}\right)\right), \end{aligned} \tag{6}$$

where $\alpha$ is a scale factor. We set $\alpha$ to 3 for all experiments. The above equations indicate that more weight is given to the output of joint cross-attention for the noisier branch (relatively closer to $t = 0$), letting it follow the domain structure of the cleaner branch.

We also introduce the unbalanced timestep sampling strategy to enforce joint distribution modeling at any separate timesteps (See Fig. 3-c). Prior studies [19, 73] have investigated the impact of timestep sampling strategies on diffusion performance during training and have proposed various timestep sampling methods beyond uniform distribution. Similarly, our base training code also employs a weighted timestep distribution for training[†] ($f(t)$ in Fig 3-c). However, with this timestep distribution, the joint distribution of $t_x$ and $t_y$ is likely insufficient to fully cover both joint generation and conditional generation tasks, as shown by Hang *et al.* [23], where insufficient timestep sampling negatively affected diffusion performance. The unbalanced timestep sampling strategy samples $t_x$ and $t_y$ independently from two unbalanced timestep distributions, $f(t)$ and $g(t)$, with half probability during training. For the remaining half, the same timesteps sampled from $f(t)$ are assigned to $t_x$ and $t_y$. We experimentally validate that these two simple techniques are effective for building a solid joint distribution across all noise levels of images and depth maps, enhancing performance in joint generation, depth-conditioned image generation, and depth estimation.

## 4. Experiments

We evaluate the performance of JointDiT across joint generation, depth estimation, and depth-conditioned image gen-

[†]https://github.com/kohya-ss/sd-scripts/tree/sd3

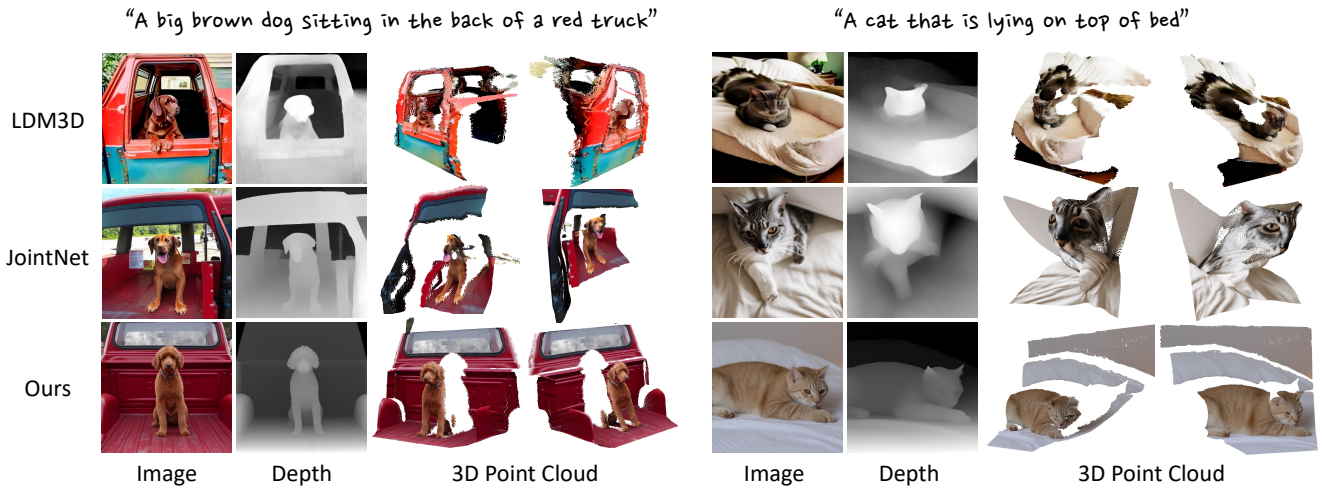


Figure 4. **3D lifting results of LDM3D [60], JointNet [71], and our JointDiT.** Our JointDiT generates highly plausible image-aligned 3D structures, surpassing previous joint generation methods in achieving superior consistency with real 3D space.

eration. We also analyze the behavior of the RGB and depth branches, as well as the effectiveness of adaptive scheduling weights and unbalanced timestep sampling. Experimental details can be found in the supplementary material.

**Implementation details.** To collect the training dataset, we randomly sample frames from a real-world internal video dataset, which allows us to acquire real-world images with a larger field of view easily. The sampled frames are resized while maintaining their aspect ratio, then center-cropped, to produce $512 \times 512$ images. The depth maps and text prompts are generated by Depth-Anything-v2 [68] and LLaVa [40], respectively. We train our model on the collected dataset, which consists of 50k pairs, for 75k iterations with a batch size of 4 and a learning rate of 1e-5. We use the LoRA rank of 64 in DiT blocks and apply text drop with a probability of 10% [25]. The training is conducted on a single NVIDIA H100 GPU for 3.5 days.

### 4.1. Joint Generation

We demonstrate JointDiT's joint generation capability through visualizations of generated images, depth maps, and their 3D lifting results, which provide intuitive evidence of joint RGB–Depth modeling and underscore the necessity of joint distribution modeling. To obtain the 3D lifting results, we apply an inverse projection to the generated image using the generated depth map. We compare our method with LDM3D [60] and JointNet [71], as they provide raw depth maps that facilitate 3D lifting visualization.

Figure 4 demonstrates the results. Compared to LDM3D and JointNet, our JointDiT shows high-fidelity images, fine-detailed depth maps, and geometrically accurate 3D lifting results. In contrast, the 3D lifting results of JointNet and LDM3D are geometrically inaccurate. We assume that this significant gap in geometric accuracy is caused by the differences in the image prior and the architecture between the baseline models, *i.e.*, stable diffusion [52] and Flux [7]. The Flux model, which is built on the diffusion transformer architecture, demonstrates superior image generation quality over stable diffusion that adopts the UNet architecture. In addition, the transformer architecture has been shown effective in depth estimation by several studies [1, 37, 50] since it has the global receptive field different from the fully-convolutional networks.

### 4.2. Depth Estimation

We assess the depth estimation capability of JointDiT with different time steps, $t_x = 1$ and $t_y = 0$. We compare our method with joint generation methods that support depth estimation, *e.g.*, JointNet [71] and UniCon [35]. We also compare with discriminative depth estimation methods, including depth-specialized models [49, 50, 68] and multi-task learning-based methods [2, 3, 42, 43], as well as generative depth estimation methods that utilize a diffusion model [21, 32]. Following the evaluation convention of prior work, we compare each method on the NYUv2 [56], ScanNet [17], KITTI [5], DIODE [64], and ETH3D [55] datasets. The evaluation metrics are Absolute Mean Relative Error (AbsRel) and $\delta_1$. Table 1 summarizes the results. Compared to joint generation methods, our model achieves superior performance across all evaluation datasets. Figure 5 shows the depth estimation results of joint generative methods on the ScanNet dataset. Compared to JointNet and UniCon, our method captures sharp edges and fine details.

We also compare our method with generative depth estimation models, which finetune most of the parameters of a pre-trained diffusion model. Except for the ETH3D dataset, our model achieves comparable performance with only a

| Type | Method | NYUv2 [56] | | ScanNet [17] | | KITTI [5] | | DIODE [64] | | ETH3D [55] | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | AbsRel ↓ | $\delta 1$ ↑ | AbsRel ↓ | $\delta 1$ ↑ | AbsRel ↓ | $\delta 1$ ↑ | AbsRel ↓ | $\delta 1$ ↑ | AbsRel ↓ | $\delta 1$ ↑ |
| Discriminative depth estimation | MiDaS [49] | 11.1 | 88.5 | 12.1 | 84.6 | 23.6 | 63.0 | 33.2 | 71.5 | 18.4 | 75.2 |
| | DPT [50] | 9.8 | 90.3 | 8.2 | 93.4 | 10.0 | 90.1 | 18.2 | 75.8 | 7.8 | 94.6 |
| | Depth-Anything-V2 [68] | 4.4 | 97.9 | 4.1 | 97.9 | 7.5 | 94.8 | 6.5 | 95.4 | 13.2 | 86.2 |
| | 4M-21 [3] | 11.8 | 88.7 | 10.6 | 89.1 | 15.6 | 78.9 | 32.1 | 75.1 | 8.4 | 93.8 |
| | MultiMAE [2] | 9.2 | 91.6 | 8.7 | 92.3 | 16.9 | 75.1 | 35.2 | 71.9 | 10.6 | 89.9 |
| | Unified-IO [42] | 6.8 | 95.9 | 7.5 | 95.0 | 28.1 | 52.0 | 36.4 | 70.0 | 13.9 | 83.9 |
| | Unified-IO 2 [43] | 12.5 | 85.8 | 14.5 | 81.6 | 48.9 | 31.7 | 43.4 | 62.0 | 20.5 | 72.1 |
| Generative depth estimation | Marigold [32] | 5.5 | 96.4 | 6.4 | 95.1 | 9.9 | 91.6 | 30.8 | 77.3 | 6.5 | 96.0 |
| | GeoWizard [21] | 5.2 | 96.6 | 6.1 | 95.3 | **9.7** | **92.1** | 29.7 | **79.2** | **6.4** | 96.1 |
| Generative joint generation | JointNet [71] | 13.7 | 81.9 | 14.7 | 79.5 | 20.9 | 66.7 | 35.0 | 58.5 | 27.1 | 73.5 |
| | UniCon [35] | 7.9 | 93.9 | 9.2 | 91.9 | — | — | — | — | — | — |
| | Ours | 5.7 | 96.9 | 6.6 | 95.7 | 10.3 | 88.8 | 27.3 | 71.0 | 16.5 | 96.3 |
| | Ours+ft | **5.0** | **97.3** | **5.6** | **96.5** | 10.9 | 87.7 | **26.6** | 71.1 | 9.3 | **96.8** |

Table 1. **Depth estimation results.** We compare ours with generative joint generation methods, as well as with discriminative and generative depth estimation methods. Ours outperforms JointNet and UniCon. Additionally, it achieves comparable performance to generative depth estimation methods, except on ETH3D. **Bold** indicates the best performance among the generative methods in this table.

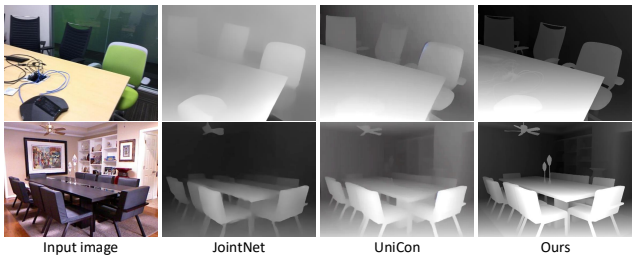


Figure 5. **Depth estimation results of joint generation models.** We visualize the depth estimation results of JointNet, UniCon, and our method on the NYUv2 and ScanNet dataset. Our approach captures thin and fine-grained details with only a timestep adjustment, *i.e.*, $t_x = 1$ and $t_y = 0$. In contrast, JointNet requires additional fine-tuning for depth estimation.

small portion of parameter tuning, *e.g.*, LoRA layers and the joint connection module. On the ETH3D dataset, our method appears to achieve higher AbsRel than generative depth estimation methods, likely because we use the depth predictions of Depth-Anything-V2 for training.

To verify the depth estimation performance itself, we further trained our model for an additional 50k iterations on synthetic datasets. We collect the synthetic training dataset by filtering 80k data samples from Hypersim [51], Replica [30], IRS [66], and MatrixCity [36]. The training method remains the same as before. As shown in Tab 1, the fine-tuned model, denoted as Ours+ft, achieved higher accuracy on the NYUv2, ScanNet, and DIODE datasets compared to generative depth estimation models. These results suggest two aspects. First, the strong image prior and architectural properties of the diffusion transformer are effective for dense prediction tasks, even with only a small subset of trainable parameters. Second, solid joint distribution modeling can serve as an alternative to conditional generation.

| Method | OpenImages 6K | |
|---|---|---|
| | FID↓ | AbsRel ↓ |
| Readout-Guidance [45] | 18.72 | 23.19 |
| ControlNet [72] | 13.68 | 9.85 |
| UniCon [35] | 13.21 | 9.26 |
| Ours | **12.62** | **6.99** |

Table 2. **Depth-conditioned image generation results.** With the same training dataset, ours achieves the lowest FID and AbsRel.

### 4.3. Depth-Conditioned Image Generation

We validate the depth-conditioned image generation quality, another joint generation with different time steps, $t_x = 0$ and $t_y = 1$. Following the evaluation protocol of UniCon [35], We compare our method with Readout-Guidance [45], ControlNet [72], and UniCon. All methods are trained on the same dataset, *i.e.*, 16k samples from PascalVOC [20], and evaluated on 6k samples from OpenImages [33]. The evaluation is based on the FID score between the generated and original images, as well as the consistency of depth estimation results, *e.g.*, AbsRel. Table 2 shows the depth-conditioned image generation results on the OpenImages 6K dataset. Compared to other methods, our method shows a lower FID score and AbsRel. The lower AbsRel indicates that the generated images accurately preserve the original image's geometry.

### 4.4. Ablation Studies

**Joint RGB-Depth feature visualization.** As shown in Fig. 2, joint RGB-Depth modeling enables plausible depth

| Adaptive scheduling weights | Unbalanced timestep | ImageNet 6K | | | Pexels 6K | | | MSCOCO 30K | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | FID↓ | IS↑ | CLIP↑ | FID↓ | IS↑ | CLIP↑ | FID↓ | IS↑ | CLIP↑ |
| ✗ | ✗ | 30.88 | 31.61 | 29.80 | 21.85 | 20.02 | 30.53 | 15.17 | 29.73 | 30.21 |
| ✗ | ✓ | 29.37 | 33.36 | 29.89 | 22.01 | 19.68 | 30.15 | 13.76 | 30.73 | 30.43 |
| ✓ | ✗ | **24.20** | 37.04 | 30.37 | **19.49** | 21.60 | 30.53 | **11.13** | 33.98 | 30.63 |
| ✓ | ✓ | 24.26 | **37.81** | **30.51** | 19.87 | **22.51** | **30.71** | 11.27 | **34.35** | **30.76** |

Table 3. **Ablation studies on joint generation.** Applying adaptive scheduling weights notably improves all evaluation metrics across all datasets. The unbalanced timestep sampling strategy enhances IS and CLIP scores when combined with adaptive scheduling weights.

| Adaptive scheduling weights | Unbalanced timestep | NYUv2 | | ScanNet | | OpenImages 6K | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | AbsRel ↓ | $\delta_1$ ↑ | AbsRel ↓ | $\delta_1$ ↑ | FID↓ | ImageReward | | | |
| | | | | | | | Rank1↑ | Rank2 | Rank3 | Rank4↓ |
| ✗ | ✗ | 8.8 | 92.2 | 10.4 | 88.9 | **11.94** | 26.35 | 26.38 | 24.43 | 22.83 |
| ✗ | ✓ | 7.8 | 93.9 | 8.7 | 92.4 | 12.51 | 21.77 | 24.82 | 25.48 | 27.93 |
| ✓ | ✗ | 6.4 | 96.0 | 7.2 | 94.9 | 14.37 | 21.15 | 22.73 | 25.63 | 30.48 |
| ✓ | ✓ | **5.7** | **96.9** | **6.6** | **95.7** | 12.58 | **30.73** | 26.07 | 24.45 | **18.75** |

Table 4. **Ablation studies on depth estimation and depth-conditioned image generation.** Adaptive scheduling weights and unbalanced timestep sampling are effective for depth estimation. In depth-conditioned image generation, using both methods together achieved the best performance in ImageReward [67] ranking, trained to capture human preference, with the first rank highest and the last ranking lowest.

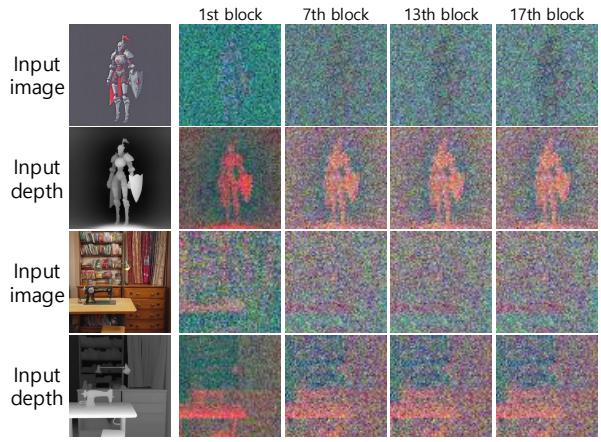


Figure 6. **Feature visualization of RGB-Depth branches in MM-DiT blocks.** We observe that, at timestep $t = 0.48$, the depth branch tends to focus on scene geometry, while the RGB branch captures semantic patterns related to texture and appearance.

generation from text prompts, even for stylized domains such as cartoon-style and pixel art illustrations. Furthermore, it tends to yield more structured and plausible 3D lifting results, not only for stylized domains but also for realistic image domains, compared to depth prediction from RGB images alone, as can be found in the supplementary material. The way the RGB and depth branches focus on textures and structure, respectively, may hint at how such capability achieves. Figure 6 shows the joint RGB-Depth features of MM-DiT blocks, visualized following the method of Tumanyan *et al*. [63]. The depth branch focuses on capturing underlying structural information, while the RGB branch attends to complementary aspects related to texture and appearance. This behavior may suggest that the depth branch, by focusing on geometric properties throughout the joint generation process, enables plausible depth generation even in scenes where depth estimation is challenging.

**Adaptive scheduling weights and unbalanced timestep sampling.** We analyze the effectiveness of adaptive scheduling weights and unbalanced timestep sampling in achieving solid joint distribution modeling. For comparison, we train four models on our dataset for 75k iterations, varying the use of adaptive scheduling weights and unbalanced timestep sampling by either applying or omitting. When not using unbalanced timestep sampling, we respectively sample $t_x$ and $t_y$ from timestep distribution $f(t)$, which is the distribution that our base training code suggests. We first investigate the effect in joint generation by assessing the quality of images generated from text prompts on three datasets: 30k samples from MS-COCO [38], 6k from ImageNet [53], and 6k from Pexels [47]. As evaluation metrics, we use the Inception Score (IS) [54], Fréchet Inception Distance (FID) [24], and CLIP similarity [48] as evaluation metrics. Table 3 shows that the usage of adaptive scheduling weights significantly improves all metrics across all evaluation datasets. When unbalanced timestep sampling is applied together, IS and CLIP scores tend to improve. These results suggest that considering the relative

noise level is important for effectively connecting different modality generation branches in separate timestep training.

We also evaluate each model on depth estimation and depth-conditioned image generation to assess the joint distribution modeling performance at the most extreme timesteps, specifically at $t_x = 0$ and $t_y = 1$, and vice versa. We use the NYUv2, ScanNet, and OpenImages 6K datasets to evaluate the performance of depth estimation and depth-conditioned image generation. Table 4 reports the results. In depth estimation, applying either adaptive scheduling weights or unbalanced timestep sampling improves performance. The best results are achieved when both are used together. For the depth-conditioned image generation, we follow UniCon [35] for the evaluation setting, using 6K samples from the OpenImages dataset. We report the FID between original and depth-conditioned images, and also the percentage of samples ranked 1st to 4th by ImageReward [67], a human preference-trained reward model for text-to-image generation. As shown in Tab 4, applying only adaptive scheduling weights results in lower performance in terms of FID. However, when combined with unbalanced timestep sampling, the performance becomes comparable to other configurations. Notably, when adaptive scheduling weights and unbalanced timestep sampling are applied together, ImageReward ranking 1 has the highest proportion, while the last ranking has the lowest. These results demonstrate that the two techniques effectively model the joint distribution at extreme timesteps.

## 5. Related Work

**Text-to-image diffusion models.** The success of DDPM [26] has demonstrated the effectiveness of diffusion models for text-to-image generation, utilizing a forward and backward process formulated as a Markov chain. Score-based generative models [57, 58] provided another perspective by modeling the diffusion process as learning the score, *i.e.*, the gradient of the log probability density, from noisy data. It was further extended with Stochastic Differential Equations (SDEs), which unify the forward and backward processes in a continuous-time framework [59]. More recently, Flow Matching [39] has been introduced as an alternative to diffusion models, enabling exact likelihood training through Continuous Normalizing Flows (CNFs) [11].

Stable Diffusion [52] improved the efficiency of the diffusion process by operating in a latent space instead of the image space, allowing for a more compact and expressive representation. This approach demonstrated impressive results. While early diffusion models primarily relied on U-Net architectures, recent studies have shown that the transformer-based architecture [65] can also be highly effective for diffusion models. The diffusion transformer [46] benefits from the global receptive field and scalability of the transformers, leading to improved generation quality. Models such as Flux and PixArt-$\alpha$ [10] further demonstrate these advantages, highlighting the potential of transformers in text-to-image generation.

**Joint and conditional diffusion models.** Text-to-image diffusion models have demonstrated their effectiveness in conditional and joint generation tasks. ControlNet [72] introduced an additional zero-initialized network, enabling fine-grained control over conditional generation tasks such as depth-to-image and pose-to-image synthesis. Building on this, LooseControl [6] proposed a more relaxed conditioning approach, allowing for weaker or more flexible integration of conditional information. Meanwhile, the image prior of pre-trained diffusion models can be beneficial in a wide range of computer vision tasks, such as sound-to-image generation [61, 62], single-image 3D reconstruction [41, 70], and 3D object texturing [9, 69].

In depth-related tasks, prior works [21, 22, 32] have demonstrated the effectiveness of diffusion priors for depth estimation. Hyoseok *et al*. [28] further showed that such priors can be used to solve inverse problems, specifically depth completion. Models such as JointNet [71], LDM3D [60], and UniCon [35] were designed to model the joint distribution between images and depth maps using diffusion models. However, these models are based on a U-Net based diffusion architecture, which has a limited receptive field. This is in contrast to recent findings suggesting that diffusion transformers provide a stronger image prior and a global receptive field, which is particularly useful for dense prediction tasks [1, 37, 50]. In this paper, we leverage the advantages of diffusion transformer to model the solid joint distribution between image and depth.

## 6. Conclusion

We propose JointDiT, which models solid joint distribution. By harnessing the strong image prior and global receptive property of a state-of-the-art diffusion transformer, we build a unified model capturing multi-modal joint distribution at any separate noise levels. To achieve solid distribution, we present two simple yet effective techniques, called the adaptive scheduling weights dependent on the noise levels of modalities and unbalanced timestep sampling strategy. Through comprehensive experiments, we show that these two techniques notably improve the performance of joint generation, depth estimation, and depth-to-image generation. Our complete model generates images and depth maps, which form highly plausible and image-aligned 3D structures when lifted into 3D space. Furthermore, JointDiT achieves depth estimation performance comparable to that of diffusion-based depth estimation models, demonstrating that a joint distribution model can serve as a viable alternative for conditional distribution models.

**Limitation.** To generate an image and its corresponding depth map, JointDiT requires an input batch size of 2 and additional parameters, resulting in a 19.8% increase in network parameters and a 2.9× longer sampling time for 20 steps. Exploring adaptations to a lightweight diffusion transformer model would be a promising research direction.

## Acknowledgment

This work was supported by the MSIT (Ministry of Science, ICT), Korea, under the Global Research Support Program in the Digital Field program (RS-2024-00436680) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation). It was also supported by the KAIST cross-generation collaborative lab project, the Ministry of Science and ICT and NIPA (HPC Support Project), and Microsoft Research Asia.

## References

[1] Ashutosh Agarwal and Chetan Arora. Depthformer: Multiscale vision transformer for monocular depth estimation with global local information fusion. In *2022 IEEE international conference on image processing (ICIP)*, pages 3873–3877. IEEE, 2022. 5, 8

[2] Roman Bachmann, David Mizrahi, Andrei Atanov, and Amir Zamir. Multimae: Multi-modal multi-task masked autoencoders. In *European Conference on Computer Vision*, pages 348–367. Springer, 2022. 5, 6

[3] Roman Bachmann, Oğuzhan F Kar, David Mizrahi, Ali Garjani, Mingfei Gao, David Griffiths, Jiaming Hu, Afshin Dehghan, and Amir Zamir. 4m-21: An any-to-any vision model for tens of tasks and modalities. *Advances in Neural Information Processing Systems*, 37:61872–61911, 2024. 5, 6

[4] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. 2023. 4

[5] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9297–9307, 2019. 5, 6, 1

[6] Shariq Farooq Bhat, Niloy Mitra, and Peter Wonka. Loosecontrol: Lifting controlnet for generalized depth conditioning. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 2, 8

[7] Black Forest Labs. Flux.1. https://huggingface.co/black-forest-labs/FLUX, 2024. 1-dev. 1, 2, 3, 4, 5

[8] Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2025. 3

[9] Dave Zhenyu Chen, Yawar Siddiqui, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. Text2tex: Text-driven texture synthesis via diffusion models. In *IEEE International Conference on Computer Vision (ICCV)*, 2023. 8

[10] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-$\alpha$: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023. 8

[11] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 3, 8

[12] Hyungjin Chung and Jong Chul Ye. Deep diffusion image prior for efficient ood adaptation in 3d inverse problems. In *European Conference on Computer Vision*, pages 432–455. Springer, 2024. 1

[13] Hyungjin Chung, Byeongsu Sim, Dohoon Ryu, and Jong Chul Ye. Improving diffusion models for inverse problems using manifold constraints. *Advances in Neural Information Processing Systems*, 35:25683–25696, 2022.

[14] Hyungjin Chung, Dohoon Ryu, Michael T Mccann, Marc L Klasky, and Jong Chul Ye. Solving 3d inverse problems using pre-trained 2d diffusion models. ieee. In *CVF Conference on Computer Vision and Pattern Recognition*, pages 10–12, 2023. 1

[15] Ciprian Corneanu, Raghudeep Gadde, and Aleix M Martinez. Latentpaint: Image inpainting in latent space with diffusion models. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 4334–4343, 2024. 1

[16] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022. 1

[17] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 5, 6, 1, 3, 4, 9

[18] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In *International Conference on Machine Learning*, pages 7480–7512. PMLR, 2023. 3, 4, 2

[19] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning (ICML)*, 2024. 3, 4, 2

[20] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88:303–338, 2010. 6, 1

[21] Xiao Fu, Wei Yin, Mu Hu, Kaixuan Wang, Yuexin Ma, Ping Tan, Shaojie Shen, Dahua Lin, and Xiaoxiao Long. Geowiz-

ard: Unleashing the diffusion priors for 3d geometry estimation from a single image. In *European Conference on Computer Vision*, pages 241–258. Springer, 2024. 2, 5, 6, 8, 1

[22] Ming Gui, Johannes S. Fischer, Ulrich Prestel, Pingchuan Ma, Dmytro Kotovenko, Olga Grebenkova, Stefan Andreas Baumann, Vincent Tao Hu, and Björn Ommer. Depthfm: Fast monocular depth estimation with flow matching. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2025. 2, 8

[23] Tiankai Hang, Shuyang Gu, Chen Li, Jianmin Bao, Dong Chen, Han Hu, Xin Geng, and Baining Guo. Efficient diffusion training via min-snr weighting strategy. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7441–7451, 2023. 4

[24] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 7, 2

[25] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 5

[26] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 8

[27] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022. 3, 4, 2

[28] Lee Hyoseok, Kyeong Seon Kim, Kwon Byung-Ki, and Tae-Hyun Oh. Zero-shot depth completion via test-time alignment with affine-invariant depth prior. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3877–3885, 2025. 1, 8

[29] Álvaro Barbero Jiménez. Mixture of diffusers for scene composition and high resolution image generation. *arXiv preprint arXiv:2302.02412*, 2023. 4

[30] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Dynamicstereo: Consistent dynamic depth from stereo videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13229–13239, 2023. 6, 2

[31] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6007–6017, 2023. 1

[32] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9492–9502, 2024. 2, 5, 6, 8, 1, 3

[33] Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Andreas Veit, et al. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github.com/openimages*, 2(3):18, 2017. 6, 1

[34] 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*, pages 19730–19742. PMLR, 2023. 1

[35] Xirui Li, Charles Herrmann, Kelvin CK Chan, Yinxiao Li, Deqing Sun, Chao Ma, and Ming-Hsuan Yang. A simple approach to unifying diffusion-based conditional generation. In *International Conference on Learning Representations (ICLR)*, 2025. 2, 3, 5, 6, 8, 1, 4

[36] Yixuan Li, Lihan Jiang, Linning Xu, Yuanbo Xiangli, Zhenzhi Wang, Dahua Lin, and Bo Dai. Matrixcity: A large-scale city dataset for city-scale neural rendering and beyond. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3205–3215, 2023. 6, 2

[37] Zhaoshuo Li, Xingtong Liu, Nathan Drenkow, Andy Ding, Francis X Creighton, Russell H Taylor, and Mathias Unberath. Revisiting stereo depth estimation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6197–6206, 2021. 5, 8

[38] 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 *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, pages 740–755. Springer, 2014. 7, 4, 9

[39] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. 3, 8

[40] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 5, 2, 4

[41] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9298–9309, 2023. 8

[42] Jiasen Lu, Christopher Clark, Rowan Zellers, Roozbeh Mottaghi, and Aniruddha Kembhavi. Unified-io: A unified model for vision, language, and multi-modal tasks. *arXiv preprint arXiv:2206.08916*, 2022. 5, 6

[43] Jiasen Lu, Christopher Clark, Sangho Lee, Zichen Zhang, Savya Khosla, Ryan Marten, Derek Hoiem, and Aniruddha Kembhavi. Unified-io 2: Scaling autoregressive multimodal models with vision language audio and action. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26439–26455, 2024. 5, 6

[44] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11461–11471, 2022. 1

[45] Grace Luo, Trevor Darrell, Oliver Wang, Dan B Goldman, and Aleksander Holynski. Readout guidance: Learning con-

trol from diffusion features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8217–8227, 2024. 2, 6, 1

[46] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 8

[47] Pexels. Pexels, royalty-free stock footage website. https://www.pexels.com. Accessed: 2024-09-30. 7

[48] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 7, 2

[49] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, 44(3):1623–1637, 2020. 5, 6, 4

[50] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021. 3, 5, 6, 8

[51] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10912–10922, 2021. 6, 2

[52] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 5, 8, 2, 3

[53] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015. 7, 3

[54] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016. 7, 2

[55] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3260–3269, 2017. 5, 6, 1

[56] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V 12*, pages 746–760. Springer, 2012. 5, 6, 1, 3, 4, 9

[57] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 8

[58] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020. 8

[59] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. 8

[60] 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. 2, 5, 8, 1, 3, 4

[61] Kim Sung-Bin, Arda Senocak, Hyunwoo Ha, and Tae-Hyun Oh. Sound2vision: Generating diverse visuals from audio through cross-modal latent alignment. *arXiv preprint arXiv:2412.06209*, 2024. 8

[62] Kim Sung-Bin, Kim Jun-Seong, Junseok Ko, Yewon Kim, and Tae-Hyun Oh. Soundbrush: Sound as a brush for visual scene editing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7167–7175, 2025. 8

[63] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1921–1930, 2023. 3, 7, 1

[64] Igor Vasiljevic, Nick Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z Dai, Andrea F Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R Walter, et al. Diode: A dense indoor and outdoor depth dataset. *arXiv preprint arXiv:1908.00463*, 2019. 5, 6, 1

[65] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 8

[66] Qiang Wang, Shizhen Zheng, Qingsong Yan, Fei Deng, Kaiyong Zhao, and Xiaowen Chu. Irs: A large naturalistic indoor robotics stereo dataset to train deep models for disparity and surface normal estimation. In *2021 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2021. 6, 2

[67] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023. 7, 8, 3, 4

[68] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37:21875–21911, 2025. 5, 6, 1, 2, 3, 4

[69] Kim Youwang, Tae-Hyun Oh, and Gerard Pons-Moll. Paint-it: Text-to-texture synthesis via deep convolutional texture map optimization and physically-based rendering. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 4347–4356, 2024. 8

[70] Kim Yu-Ji, Hyunwoo Ha, Kim Youwang, Jaeheung Surh, Hyowon Ha, and Tae-Hyun Oh. Metta: Single-view to 3d textured mesh reconstruction with test-time adaptation. In *British Machine Vision Conference (BMVC)*, 2024. 8

[71] Jingyang Zhang, Shiwei Li, Yuanxun Lu, Tian Fang, David McKinnon, Yanghai Tsin, Long Quan, and Yao Yao. Jointnet: Extending text-to-image diffusion for dense distribution modeling. In *International Conference on Learning Representations (ICLR)*, 2024. 2, 5, 6, 8, 1, 3, 4

[72] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023. 2, 6, 8, 1

[73] Tianyi Zheng, Cong Geng, Peng-Tao Jiang, Ben Wan, Hao Zhang, Jinwei Chen, Jia Wang, and Bo Li. Non-uniform timestep sampling: Towards faster diffusion model training. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 7036–7045, 2024. 4

# JointDiT: Enhancing RGB-Depth Joint Modeling with Diffusion Transformers

## Supplementary Material

## Contents


**A Implementation Details**
- A.1 Experiment Setup
- A.2 Data Preprocessing
- A.3 Unbalanced Timestep Sampling Strategy
- A.4 Architecture of JointDiT

**B Additional Experiments**
- B.1 Advantages of Joint RGB-Depth Modeling
- B.2 Jointly Generated Image Quality
- B.3 Ablation of the LoRA's Rank
- B.4 Analysis of Failure Cases
- B.5 Joint Panorama Generation

**C Additional Qualitative Results**


## A. Implementation Details

We provide the details of the experiment setup, dataset preprocessing, proposed unbalanced timestep sampling strategy, and architecture design of JointDiT.

### A.1. Experiment Setup

We will describe in detail the configurations we used for joint generation, depth estimation, depth-conditioned image generation, and Joint RGB-Depth feature visualization. We consistently use 20 denoising steps across all experiments.

**Joint generation.** We generate images and their corresponding depth maps by initially setting $t_x = 0$ and $t_y = 0$ by sampling noises from a standard normal distribution. While the main paper presents joint generation results conditioned on text prompts, we find that joint generation occurs even without a text prompt. To compare with JointNet [71] and LDM3D [60], we generate 512×512 images and depth maps jointly. Despite being trained only on a 512×512 resolution dataset, we observe that JointDiT successfully operates at varying resolutions, such as 1024×1024.

**Depth estimation.** To estimate the depth map from a given image, we set $t_x = 1$ and $t_y = 0$ and provide an empty text prompt. Unlike Marigold [32] and Geowizard [21], we do not use any ensemble technique. Since JointDiT can operate at varying resolutions, we use the NYUv2, ScanNet, KITTI, and DIODE datasets [5, 17, 56, 64] at their original resolutions as model inputs. For the ETH3D dataset [55], which has a 4K resolution, we resize the images while preserving the aspect ratio so that the larger dimension is set to 1024 pixels. This preprocessing strategy is consistently applied to the comparison methods as well, and for methods that require a fixed input resolution, we use their designated resolution for evaluation.

| Type | LoRA applied components |
|---|---|
| MM-DiT | img_mod.lin<br>img_attn.qkv<br>txt_mod.lin<br>txt_attn.qkv<br>img_attn.proj<br>txt_attn.proj |
| P-DiT | linear1<br>modulation.lin |
| Input stage | vector_in.in_layer<br>vector_in.out_layer<br>txt_in |

Table 5. **LoRA-applied components.** To build the depth branch extending the original Flux model [7], we add LoRAs to MM-DiT, P-DiT, and Input stage.

**Depth-conditioned image generation.** We generate depth-conditioned images from given text prompts by initially setting $t_x = 0$ and $t_y = 1$. The conditioning depth maps are obtained by Depth-Anything-V2 [68]. For the experiment of Sec. 4.3 in the main paper, we follow the evaluation setting of UniCon [35] to compare with Readout-Guidance [45], ControlNet [72], and UniCon. Specifically, we train our model and these methods on the same training dataset, which includes 16k images of PascalVOC [20], depth maps from Depth-Anything-V2 [68], and text prompts extracted using BLIP2 [34]. For the evaluation, using the selected 6k images from the OpenImages dataset [33], we estimate depth maps using an off-the-shelf model and generate images conditioned on these depth maps and text prompts from BLIP2.

**Joint RGB-Depth feature visualization.** For the feature visualization of Sec. 4.4 in the main paper, we strictly follow the method proposed by Tumanyan *et al.* [63], and visualize the PCA results of the features from each MM-DiT block. Similar to Tumanyan *et al.*, who collected images from semantically related domains (such as humanoid pictures) for visualization, we perform joint generation on 50 samples for each domain, *i.e.*, pixel art style illustrations and indoor scenes that are used in the two examples shown in Fig. 6 of the main paper. We extract features at approximately 50% of the generation process (*i.e.*, $t = 0.48$), and

apply PCA to visualize them. Due to the architecture structure of the Flux model, which applies positional encoding immediately before every attention layer, we subsample the even indices before applying PCA.

### A.2. Data Preprocessing

We randomly sample RGB frames from the internal video dataset, which has a resolution of 512×512 or higher. The sampled frames are resized so that the smaller dimension (width or height) is 512 pixels, followed by a 512×512 center crop. We obtain text prompts from the 512×512 images using LLaVA [40]. To generate the corresponding disparity maps, we use Depth-Anything-V2 and normalize them so that the maximum and minimum values are 1 and 0, respectively.

**Synthetic dataset.** We further fine-tune our model to verify the depth estimation capability itself. We utilize the Hypersim [51], Replica [30], IRS [66], and MatrixCity [36] datasets for fine-tuning. We first unify the ground-truth depth or disparity maps of the synthetic datasets into disparity maps because our model was previously trained on the disparity maps of Depth-Anything-V2. Thereafter, we define invalid regions for each dataset. For example, in MatrixCity, the depth of the sky was set to the maximum value, while in Replica, there exist depth values that are closer than the camera plane. Then, we apply the bias and scale to the ground-truth disparity map so that the mean and standard deviation match those of Depth-Anything-V2's disparity estimation at valid regions. The annotations in invalid regions are replaced with Depth-Anything-V2's estimation. This process allows us to obtain annotations for invalid regions while ensuring consistency in depth map characteristics, which can vary significantly when normalized by maximum and minimum values due to dataset-specific invalid regions.

### A.3. Unbalanced Timestep Sampling Strategy

When applying the unbalanced timestep sampling strategy, the timesteps, *i.e.*, $t_x$ and $t_y$, are separately sampled from the timestep distributions $f(t)$ and $g(t)$, respectively, or vice versa. This is applied with a 50% probability during training, while for the remaining 50%, the same timestep sampled from $f(t)$ is used for both $t_x$ and $t_y$. The timestep distribution is as follows:

$$f(t) = 1 - \frac{\sigma(z) \cdot s}{1 + (s-1) \cdot \sigma(z)}, \quad \text{where } z \sim \mathcal{N}(0,1). \tag{7}$$

The $\sigma(\cdot)$ denotes the sigmoid function. In $f(t)$, which is suggested by our base training code[†], $s$ is set to 3.1582. We set $s$ to 0.25 to obtain $g(t)$.

[†]https://github.com/kohya-ss/sd-scripts/tree/sd3

### A.4. Architecture of JointDiT

To build the depth branch, we add LoRAs [27] to the original Flux architecture [7]. Specifically, we add LoRAs to the components connected before and after the attention mechanisms of the multi-modal diffusion transformer (MM-DiT) and parallel diffusion transformer (P-DiT) blocks [18, 19] that constitute Flux. Table 5 summarizes the LoRA-applied components in the MM-DiT and P-DiT blocks. We use a LoRA rank of 64 for both MM-DiT and P-DiT, and apply relatively larger ranks of 512 or 1024 to the input stage. The alpha value is set to half of the corresponding rank.

To design the joint connection module, we adopt the joint cross-attention module from UniCon [35], followed by a zero-initialized linear projection layer. The adaptive scheduling weight is applied subsequently.

## B. Additional Experiments

### B.1. Advantages of Joint RGB-Depth Modeling

As mentioned in the main paper, we observe that joint RGB-Depth generation tends to yield more plausible 3D lifting results compared to estimating depth from generated images. Figure 7 presents the 3D lifting results by showing top and side views. When using the depth generated by our JointDiT, the results exhibit more well-structured and volumetric 3D geometry than those produced by Marigold [32] and Depth-Anything-V2 [68].

Furthermore, as also discussed in the main paper, our joint generation approach enables plausible depth synthesis even in illustration domains, where depth estimation methods often struggle. Additional qualitative results are presented in Figure 12.

### B.2. Jointly Generated Image Quality

We quantitatively compare the quality of jointly generated images from JointNet [71], LDM3D [60], and our method. For evaluation, we use the dataset from Section 4.4 of the main paper, *i.e.*, ImageNet 6K, Pexels 6K, and MSCOCO 30K. We measure the Inception Score (IS) [54], Fréchet Inception Distance (FID) [24], and CLIP similarity [48] as our evaluation metrics. Table 6 summarizes the results. We also include the results of baseline diffusion models, *i.e.*, Stable diffusion [52] and Flux [7]. Interestingly, Flux achieves relatively high FID scores across all evaluation datasets despite its outstanding text-to-image generation capability. We observe that Flux often generates stylized images. Figure 8 shows samples from ImageNet 6K and the corresponding images generated by Flux. The generated samples appear surreal, which leads to a higher FID between them and the real image dataset. Our model achieves a lower FID score than Flux by learning the joint distribution of images and their corresponding depth maps on the real dataset. However, our IS score is lower than that of Flux, likely due to


Figure 7. **Comparison of 3D lifting results from our JointDiT, Marigold, and Depth-Anything-V2.** The jointly generated depth from JointDiT leads to more coherent 3D shapes and better preservation of structural details compared to the estimated depths.

| Generation modality | Method | ImageNet 6K | | | Pexels 6K | | | MSCOCO 30K | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | FID↓ | IS↑ | CLIP↑ | FID↓ | IS↑ | CLIP↑ | FID↓ | IS↑ | CLIP↑ |
| Image | SD v2.1 [52] | 23.13 | 40.49 | 31.16 | 20.53 | 24.73 | 31.37 | 15.00 | 37.13 | 31.37 |
| | Flux | 25.96 | 46.12 | 30.90 | 24.71 | 25.32 | 31.09 | 22.85 | 41.40 | 30.77 |
| Image & depth | JointNet [71] | 25.92 | 37.23 | 30.50 | 20.28 | 24.94 | 30.72 | 12.62 | 35.88 | 30.80 |
| | LDM3D [60] | 37.72 | 31.73 | 30.45 | 32.50 | 20.26 | 30.52 | 25.58 | 29.36 | 30.81 |
| | Ours | 24.26 | 37.81 | 30.51 | 19.87 | 22.51 | 30.71 | 11.27 | 34.35 | 30.76 |

Table 6. **Quantitative evaluation on jointly generated images.** We present the performance of the baseline model for comparison. Our method achieves performance comparable to JointNet, while LDM3D demonstrates relatively poor results. Compared to our base model, *i.e.*, Flux, we achieve lower FID scores but also lower IS scores, likely due to the limited size of the training dataset.


Figure 8. **Comparison between original images and images generated by Flux [7] on the ImageNet [53] 6K dataset.** Flux often generates stylized images, which leads to a higher FID between the real image dataset and the generated images.

the limited size of the training dataset.

Among the joint generation models, LDM3D shows relatively poor performance. Our method achieves comparable performance to JointNet. To further assess image generation quality, we evaluate the human preference score using ImageReward [67], a trained model that estimates human preference for given text prompts and images. We measure the human preference ranking of the images generated by the joint generation model from the same text prompt. Table 7 summarizes the percentage of each method on each evaluation dataset. Our method shows the highest rank 1 percentage and the lowest rank 3 percentage across all evaluation datasets. Compared to LDM3D, JointNet achieves moderately better performance.

### B.3. Ablation of the LoRA's Rank

We adopt a LoRA rank of 64 in the DiT blocks of our JointDiT model. To analyze the effect of the LoRA rank, we train our model with different LoRA ranks and evaluate depth estimation performance on the NYUv2 and ScanNet datasets [17, 56]. As shown in Table 8, as the LoRA rank increases, the depth estimation performance improves, achieving the best performance at the LoRA rank of 64. We did not increase the LoRA rank beyond 64 because the number of model parameters grows exponentially.

### B.4. Analysis of Failure Cases

We observe that our method shares similar limitations with depth estimation methods [32, 68], particularly in handling reflective surfaces such as mirrors. As shown in Fig. 9, both our model and depth estimation models fail to recognize mirrors as flat and planar regions.

| Method | ImageNet 6K | | | Pexels 6K | | | MSCOCO 30K | | |
|---|---|---|---|---|---|---|---|---|---|
| | ImageReward | | | ImageReward | | | ImageReward | | |
| | Rank1↑ | Rank2 | Rank3↓ | Rank1↑ | Rank2 | Rank3↓ | Rank1↑ | Rank2 | Rank3↓ |
| LDM3D [60] | 27.56 | 35.90 | 36.54 | 26.21 | 33.42 | 40.37 | 27.74 | 34.04 | 38.22 |
| JointNet [71] | 29.91 | 33.32 | 36.77 | 31.65 | 33.87 | 34.48 | 28.85 | 35.70 | 35.46 |
| Ours | **42.53** | 30.79 | **26.69** | **42.14** | 32.72 | **25.15** | **43.41** | 30.26 | **26.32** |

Table 7. **Human preference evaluation on images jointly generated by joint generation methods [60, 71] and Ours.** We assess the human preference using ImageReward [67] that was trained to estimate human preference. With both joint generation models and ours, we conduct joint generation using the same text prompts and rank the results with ImageReward, obtaining the percentage for each ranking. Our JointDiT achieved the highest rank 1 percentage and the lowest rank 3 percentage across all datasets.

| LoRA rank | NYUv2 [56] | | ScanNet [17] | |
|---|---|---|---|---|
| | AbsRel ↓ | $\delta_1$ ↑ | AbsRel ↓ | $\delta_1$ ↑ |
| 16 | 9.1 | 90.6 | 9.8 | 89.7 |
| 32 | 6.6 | 95.7 | 8.5 | 92.4 |
| 64 (Ours) | **5.7** | **96.9** | **6.6** | **95.7** |

Table 8. **Ablation studies of the rank of LoRA.** We evaluate the depth estimation performance on NYUv2 and ScanNet while varying the LoRA rank. The results show that performance improves as the LoRA rank increases.

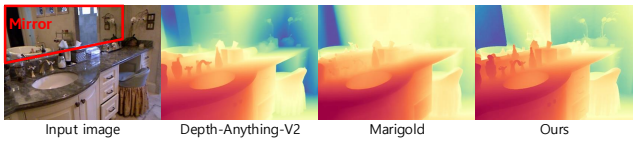


Figure 9. **Failure cases in depth estimation.** Red and Blue areas indicate near and far depth predictions, respectively.

### B.5. Joint Panorama Generation

JointDiT can be used for RGB-D panorama generation as well. For panorama generation, we strictly follow the JointNet [71] method combining whole and tile-based denoising strategies [4, 29], to ensure a fair comparison. We denoise image and depth tiles by using joint generative diffusion models. During only early steps, we perform denoising on the entire panorama, and throughout all steps, we aggregate model estimations from both overlapped individual tiles and the whole panorama. Figure 10 demonstrates the RGB-D panorama results. Compared to JointNet, JointDiT shows clear and structurally reasonable images along with sharp depth maps.

## C. Additional Qualitative Results

In this section, we demonstrate diverse qualitative results on depth estimation and depth-conditioned image generation.

**Joint generation.** Utilizing our JointDiT model, we generate images and corresponding depth maps. We visualize the images and depths with their 3D lifting results. As shown in Fig. 11, our joint generation results are geometrically reasonable in 3D, with the surface characteristics of the images being well-preserved in the 3D space (e.g., smooth or rough textures). Furthermore, Figure 12 highlights the effectiveness of our joint generation approach in illustration domains, where plausible 3D structures are obtained despite the inherent difficulty of estimating geometry from stylized images.

**Depth estimation.** We visualize the depth estimation results of joint generation methods that support depth estimation, *i.e.*, JointNet [71], UniCon [35], and Ours. We obtain the depth estimation results from the publicly available code. Specifically, while UniCon does not provide raw depth through its Gradio demo, we can obtain depth estimation visualization results. To estimate depth, we provide each model with empty text prompts. To demonstrate the results across various scenarios, we acquire depth maps estimated from the NYUv2, ScanNet, and MSCOCO datasets [17, 38, 56]. Figure 13 illustrates the results. Compared to JointNet and UniCon, our method captures fine details in the depth and the shape of thin objects. This aligns with the trends observed in the quantitative results.

**Depth-conditioned image generation.** We visualize the depth-conditioned image generation results of JointNet, UniCon, and our method. We utilize publicly available code for the other two methods. To generate the results, we obtain depth maps and text prompts from ImageNet 6K using Depth-Anything-V2 [68] and LLaVA [40]. For JointNet, we provide the depth estimation from MiDaS [49], as it was trained using MiDaS' depth estimation. Figure 14 demonstrates the results. JointNet and UniCon generally generate images that match the given depth and text prompts, but they sometimes do not fully understand the text prompt. For example, UniCon generated a green dog instead of a green frisbee, and JointNet failed to fully generate a red flower.

In comparison, our JointDiT shows generation results that are well aligned with the given depth and text prompts, and we observe that it generates more realistic images than the other models.

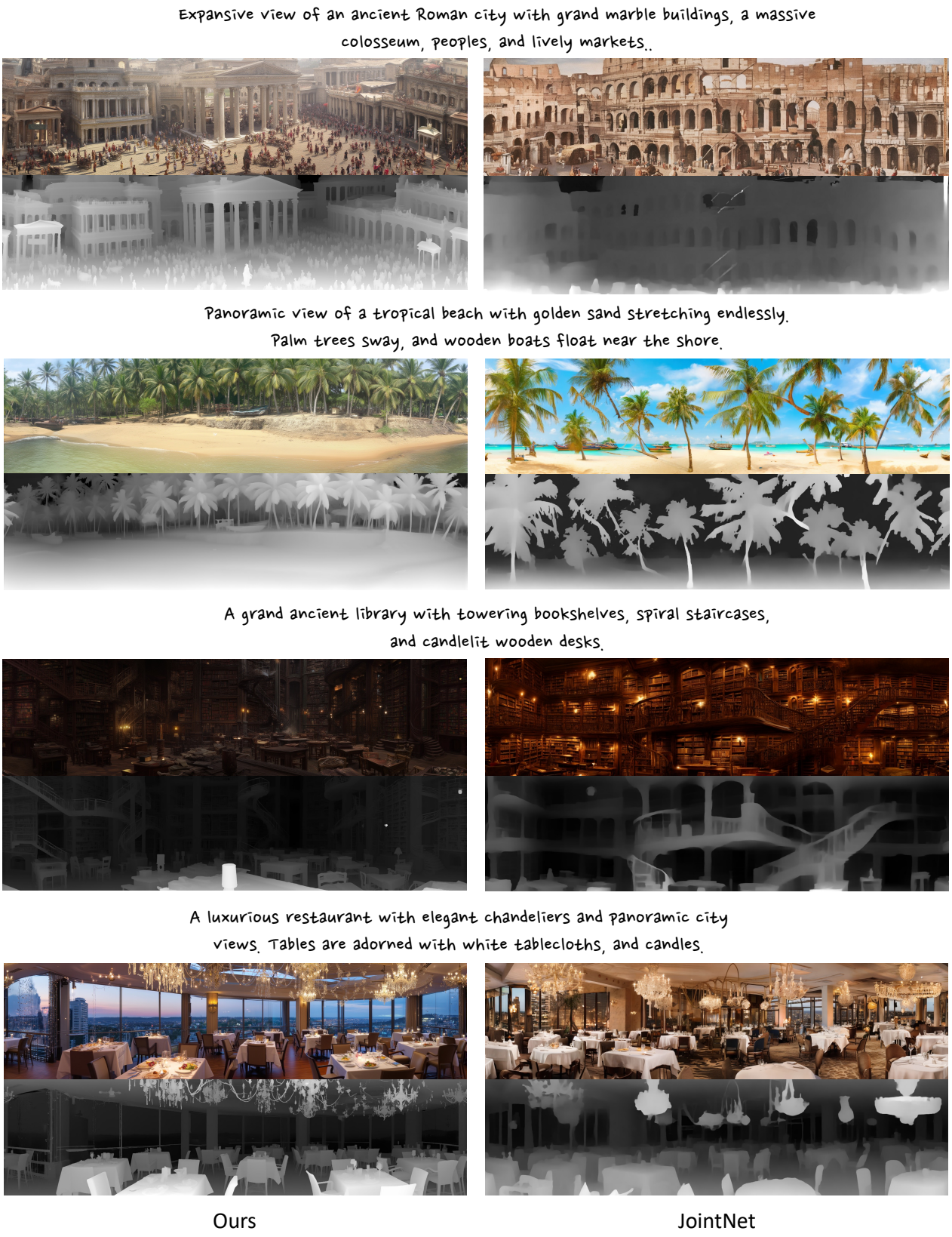


Figure 10. **RGB-D panoramic generation results of JointNet and Ours.** Our JointDiT generates more three-dimensional and sharper images and depth maps compared to JointNet.

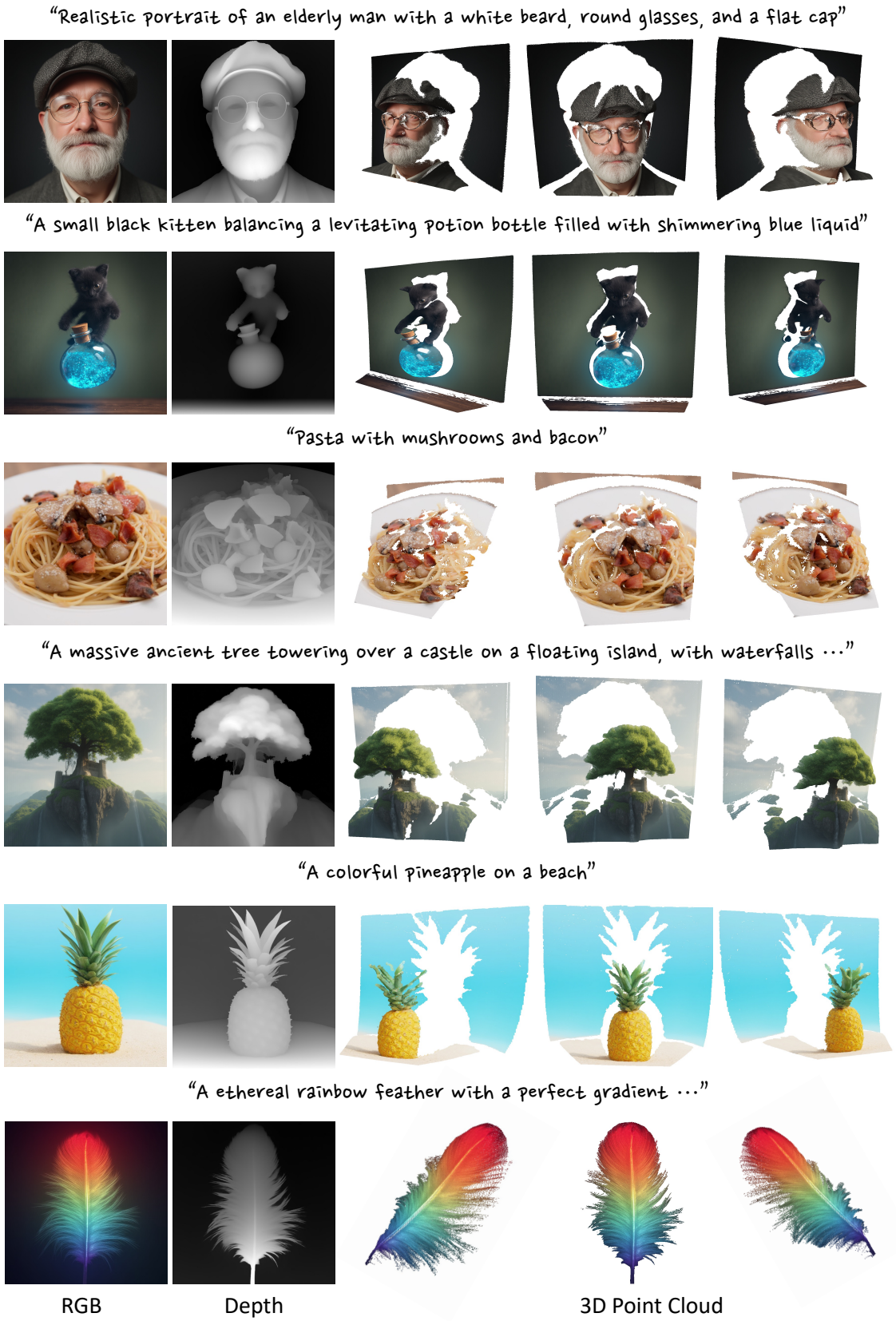


Figure 11. **Joint generation results of JointDiT.** The joint generated images and depths are geometrically reasonable in 3D.

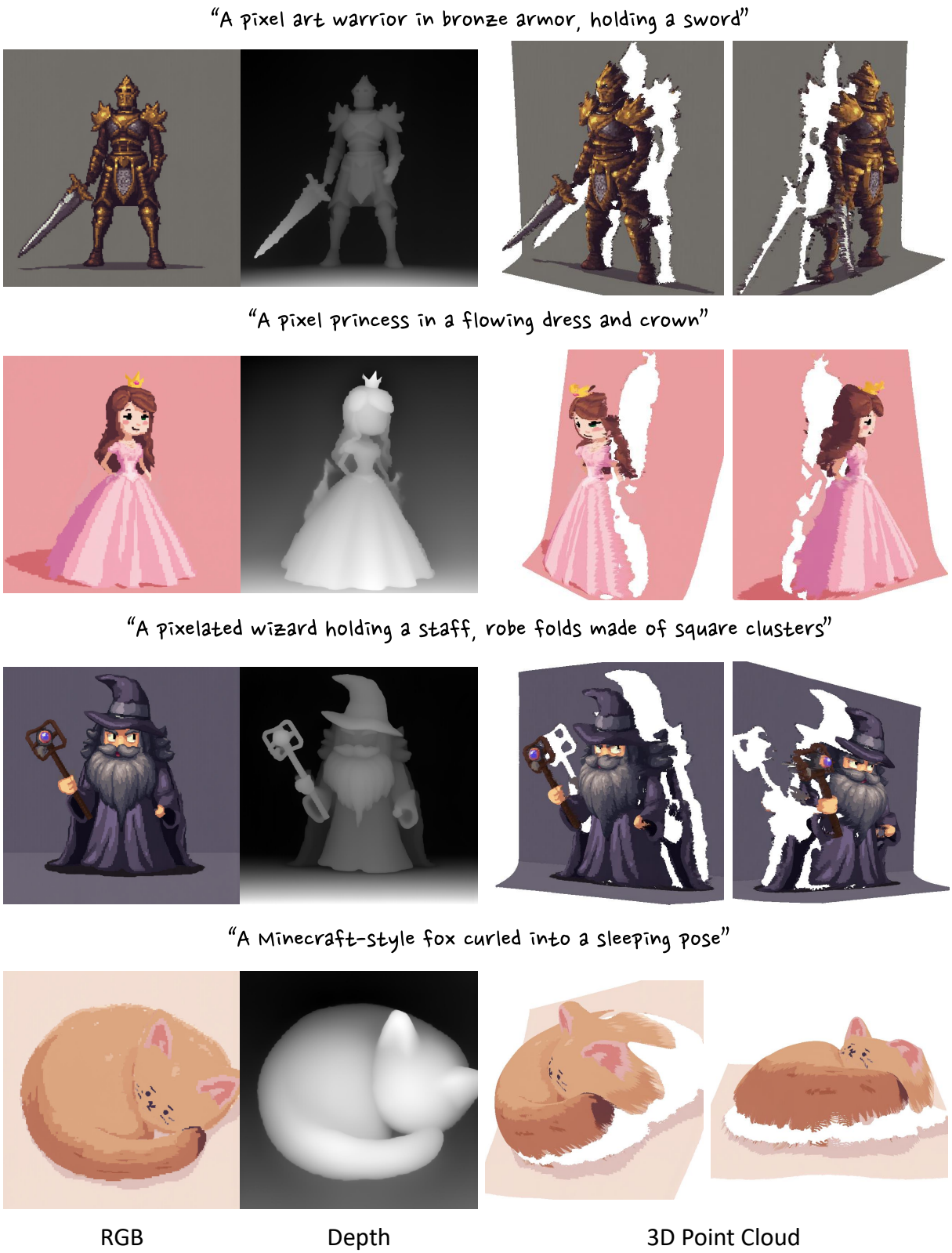


Figure 12. **Joint generation results in illustration domains.** The jointly generated images and depths from JointDiT produce geometrically plausible 3D structures, even in stylized domains.

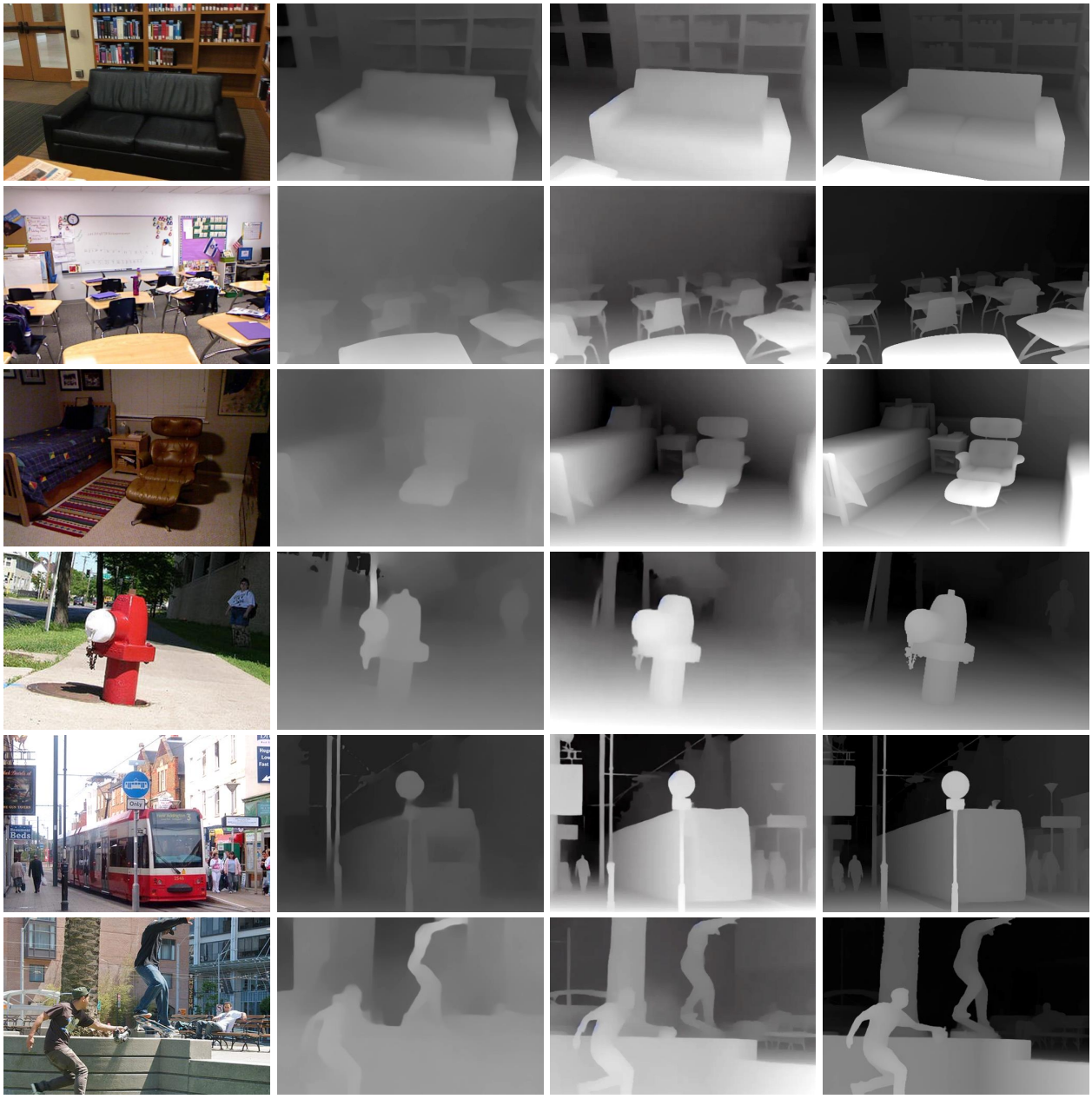


Figure 13. **Depth estimation results of joint generation models.** We visualize the depth estimation results of JointNet, UniCon, and our method on the NYUv2, ScanNet, MSCOCO dataset [17, 38, 56]. Our method shows sharp and fine-detailed depth visualization, which aligns with the trends observed in the qualitative results.


Figure 14. **Depth-conditioned image generation results of JointNet, UniCon, and Ours.** JointNet and UniCon often fail to reflect the text prompt properly, *e.g.*, the green dog generated by UniCon and the flower with green petals generated by JointNet. Our JointDiT generates images that better reflect the text prompt and depth map, producing more realistic results compared to other methods.