Supplementary Material for 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

Contents

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 [35] and LDM3D [29], 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.

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

Table S1. **LoRA-applied components.** To build the depth branch extending the original Flux model [3], we add LoRAs to MM-DiT, P-DiT, and Input stage.

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 [13] and Geowizard [8], we do not use any ensemble technique. Since JointDiT can operate at varying resolutions, we use the NYUv2, ScanNet, KITTI, and DIODE datasets [2, 4, 28, 31] at their original resolutions as model inputs. For the ETH3D dataset [27], 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.

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 [34]. For the experiment of Sec. 4.3 in the main paper, we follow the evaluation setting of UniCon [16] to compare with Readout-Guidance [20], ControlNet [36], and Uni-

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

Con. Specifically, we train our model and these methods on the same training dataset, which includes 16k images of PascalVOC [7], depth maps from Depth-Anything-V2 [34], and text prompts extracted using BLIP2 [15]. For the evaluation, using the selected 6k images from the OpenImages dataset [14], 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.* [30], 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 [19]. 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 [23], Replica [12], IRS [32], and MatrixCity [17] 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:

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

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 $g(t)$.

A.4. Architecture of JointDiT

To build the depth branch, we add LoRAs [10] to the original Flux architecture [3]. 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 [5, 6] that constitute Flux. Table S1 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 [16], 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 S1 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 [13] and Depth-Anything-V2 [34].

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 S6.

B.2. Jointly Generated Image Quality

We quantitatively compare the quality of jointly generated images from JointNet [35], LDM3D [29], and our method.

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

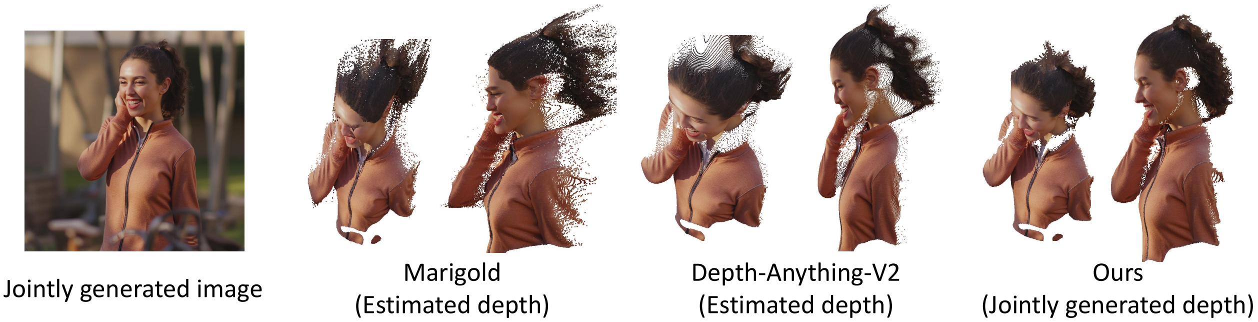

Figure S1. **Comparison of 3D lifting results from our JointDiT, Marigold, and Depth-Anything-V2.** The jointly generated depth from JointDiT leads to more coherent 3D shapes and better preservation of structural details compared to the estimated depths.

Generation modality	Method	ImageNet 6K			Pexels 6K			MSCOCO 30K		
		FID↓	IS↑	CLIP↑	FID↓	IS↑	CLIP↑	FID↓	IS↑	CLIP↑
Image	SD v2.1 [24]	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 [35]	25.92	37.23	30.50	20.28	24.94	30.72	12.62	35.88	30.80
	LDM3D [29]	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 S2. **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.

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) [26], Fréchet Inception Distance (FID) [9], and CLIP similarity [21] as our evaluation metrics. Table S2 summarizes the results. We also include the results of baseline diffusion models, *i.e.*, Stable diffusion [24] and Flux [3]. 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 S2 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 the limited size of the training dataset.

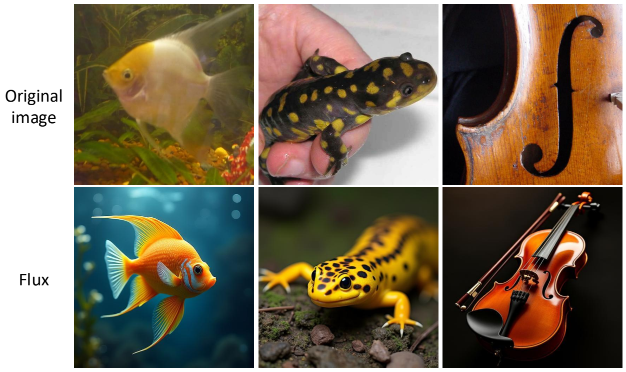

Figure S2. **Comparison between original images and images generated by Flux [3] on the ImageNet [25] 6K dataset.** Flux often generates stylized images, which leads to a higher FID between the real image dataset and the generated images.

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 [33], 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 S3 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.

Method	ImageNet 6K			Pexels 6K			MSCOCO 30K		
	ImageReward			ImageReward			ImageReward		
	Rank1↑	Rank2	Rank3↓	Rank1↑	Rank2	Rank3↓	Rank1↑	Rank2	Rank3↓
LDM3D [29]	27.56	35.90	36.54	26.21	33.42	40.37	27.74	34.04	38.22
JointNet [35]	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 S3. **Human preference evaluation on images jointly generated by joint generation methods [29, 35] and Ours.** We assess the human preference using ImageReward [33] 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.

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 [4, 28]. As shown in Table S4, 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.

LoRA rank	NYUv2 [28]		ScanNet [4]	
	AbsRel ↓	δ_1 ↑	AbsRel ↓	δ_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 S4. **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.

B.4. Analysis of Failure Cases

We observe that our method shares similar limitations with depth estimation methods [13, 34], particularly in handling reflective surfaces such as mirrors. As shown in Fig. S3, both our model and depth estimation models fail to recognize mirrors as flat and planar regions.

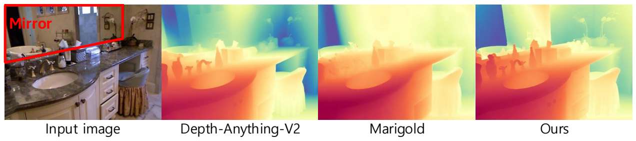

Figure S3. **Failure cases in depth estimation.** Red and Blue areas indicate near and far depth predictions, respectively.

B.5. Joint Panorama Generation

JointDiT can be used for RGB-D panorama generation as well. For panorama generation, we strictly follow the JointNet [35] method combining whole and tile-based denoising strategies [1, 11], 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 S4 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. S5, 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 S6 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 [35], UniCon [16], 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 [4, 18, 28]. Figure S7 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 [34] and LLaVA [19]. For JointNet, we provide the depth estimation from MiDaS [22], as it was trained using MiDaS' depth estimation. Figure S8 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.

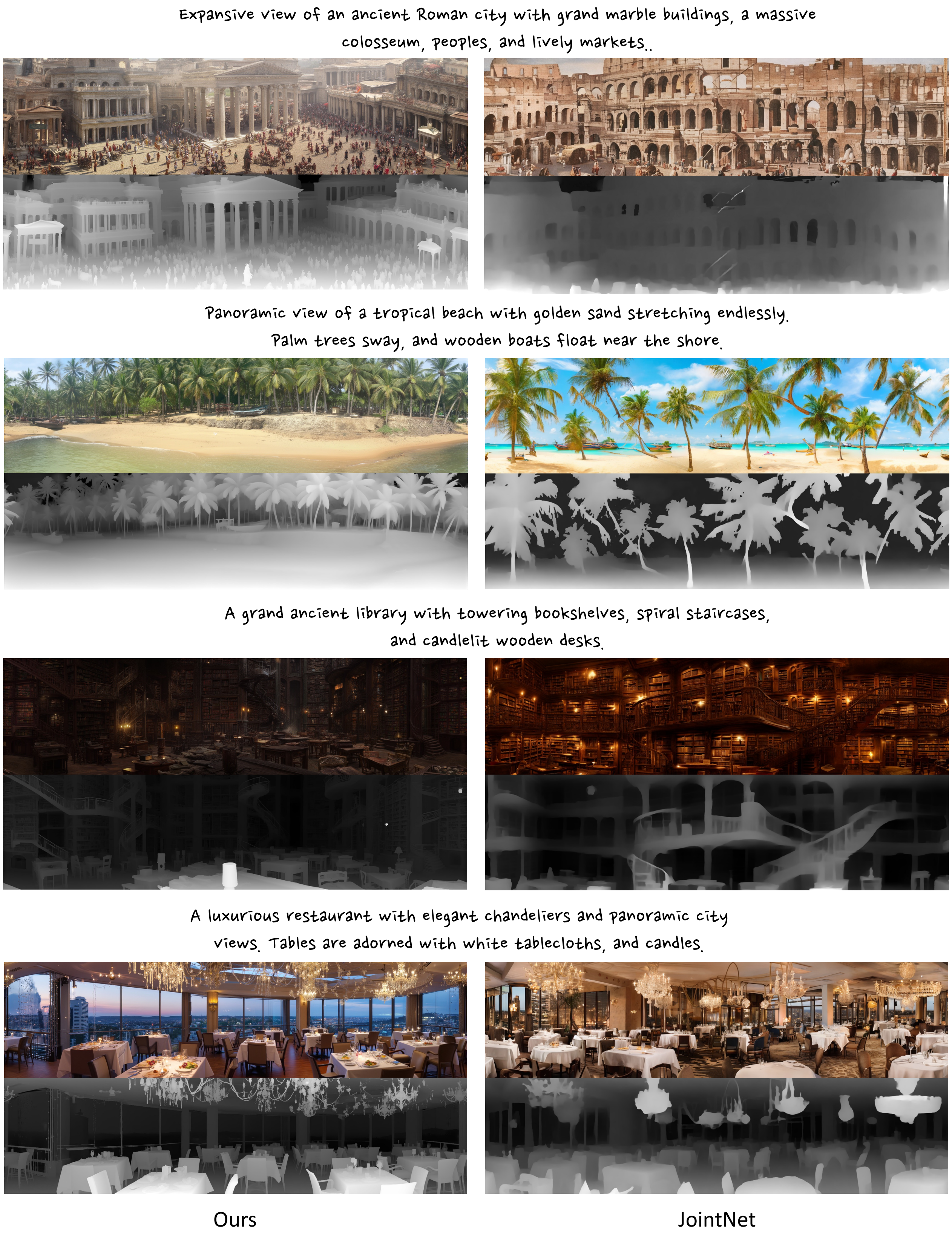
Expansive view of an ancient Roman city with grand marble buildings, a massive colosseum, peoples, and lively markets..
Panoramic view of a tropical beach with golden sand stretching endlessly. Palm trees sway, and wooden boats float near the shore.
A grand ancient library with towering bookshelves, spiral staircases, and candlelit wooden desks.
A luxurious restaurant with elegant chandeliers and panoramic city views. Tables are adorned with white tablecloths, and candles.
Ours
JointNet

Figure S4. **RGB-D panoramic generation results of JointNet and Ours.** Our JointDiT generates more three-dimensional and sharper images and depth maps compared to JointNet.

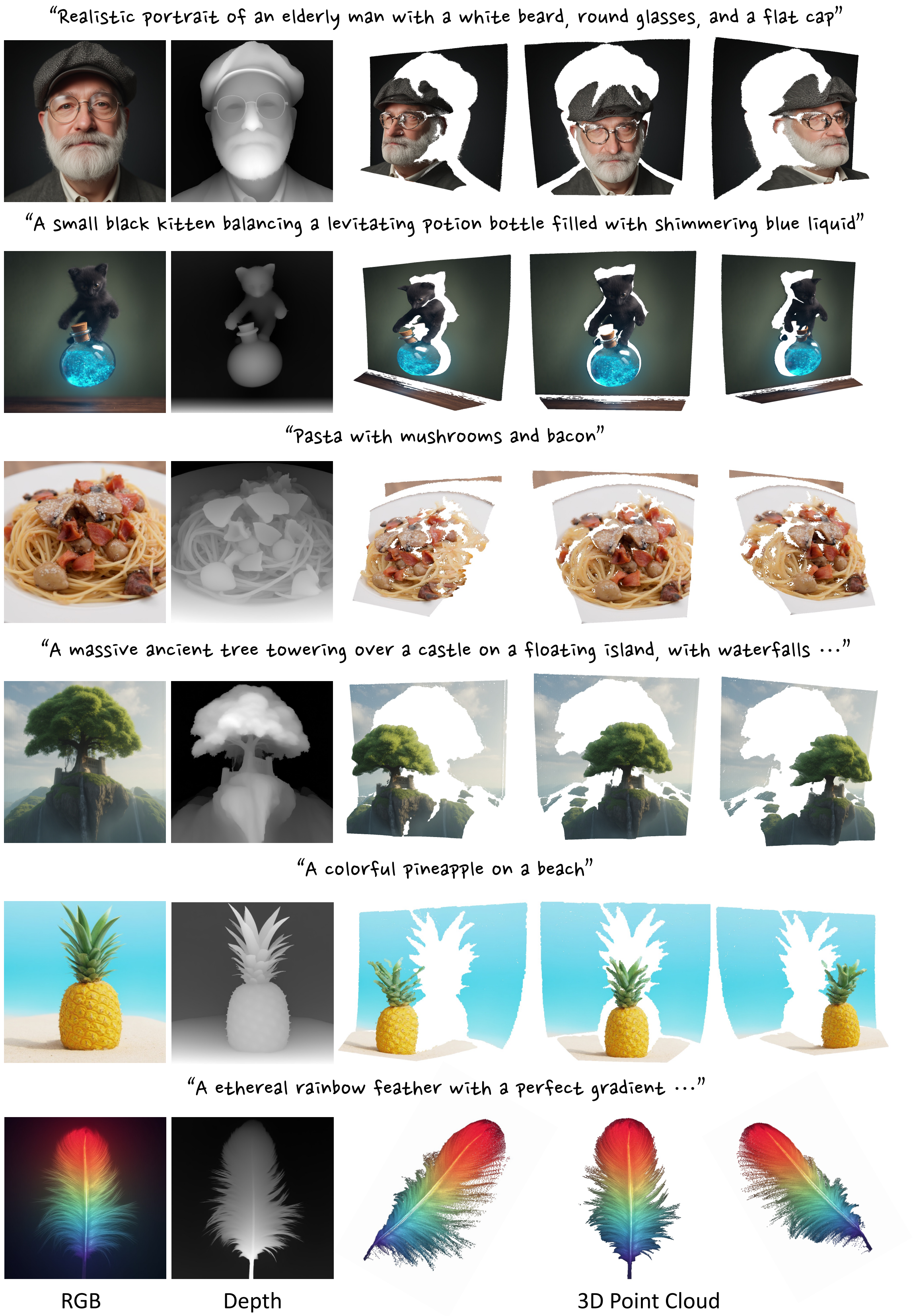
"Realistic portrait of an elderly man with a white beard, round glasses, and a flat cap"
"A small black kitten balancing a levitating potion bottle filled with shimmering blue liquid"
"Pasta with mushrooms and bacon"
"A massive ancient tree towering over a castle on a floating island, with waterfalls ..."
"A colorful pineapple on a beach"
"A ethereal rainbow feather with a perfect gradient ..."
RGB
Depth
3D Point Cloud

Figure S5. **Joint generation results of JointDiT.** The joint generated images and depths are geometrically reasonable in 3D.

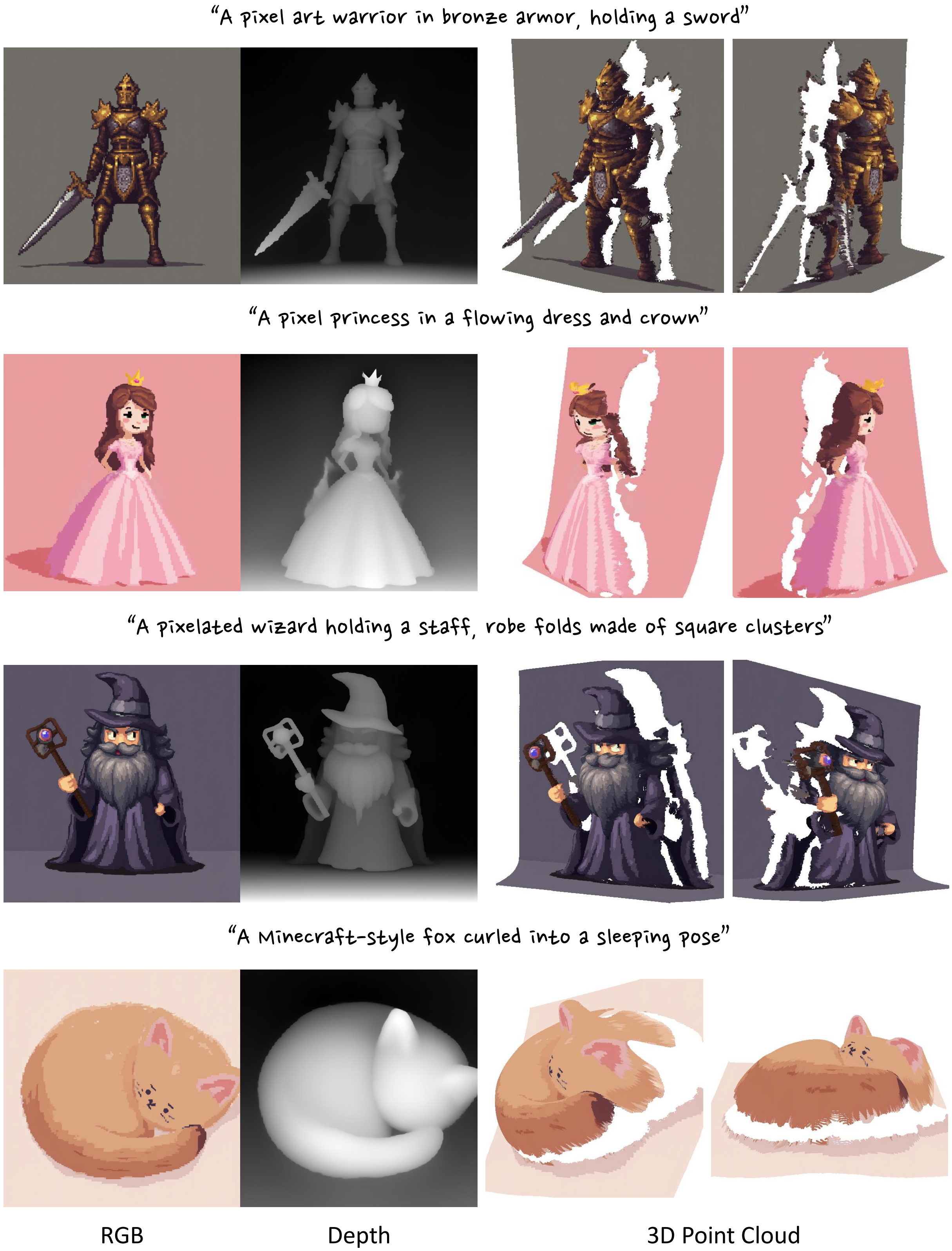
"A pixel art warrior in bronze armor, holding a sword"
"A pixel princess in a flowing dress and crown"
"A pixelated wizard holding a staff, robe folds made of square clusters"
"A Minecraft-style fox curled into a sleeping pose"
RGB
Depth
3D Point Cloud

Figure S6. **Joint generation results in illustration domains.** The jointly generated images and depths from JointDiT produce geometrically plausible 3D structures, even in stylized domains.

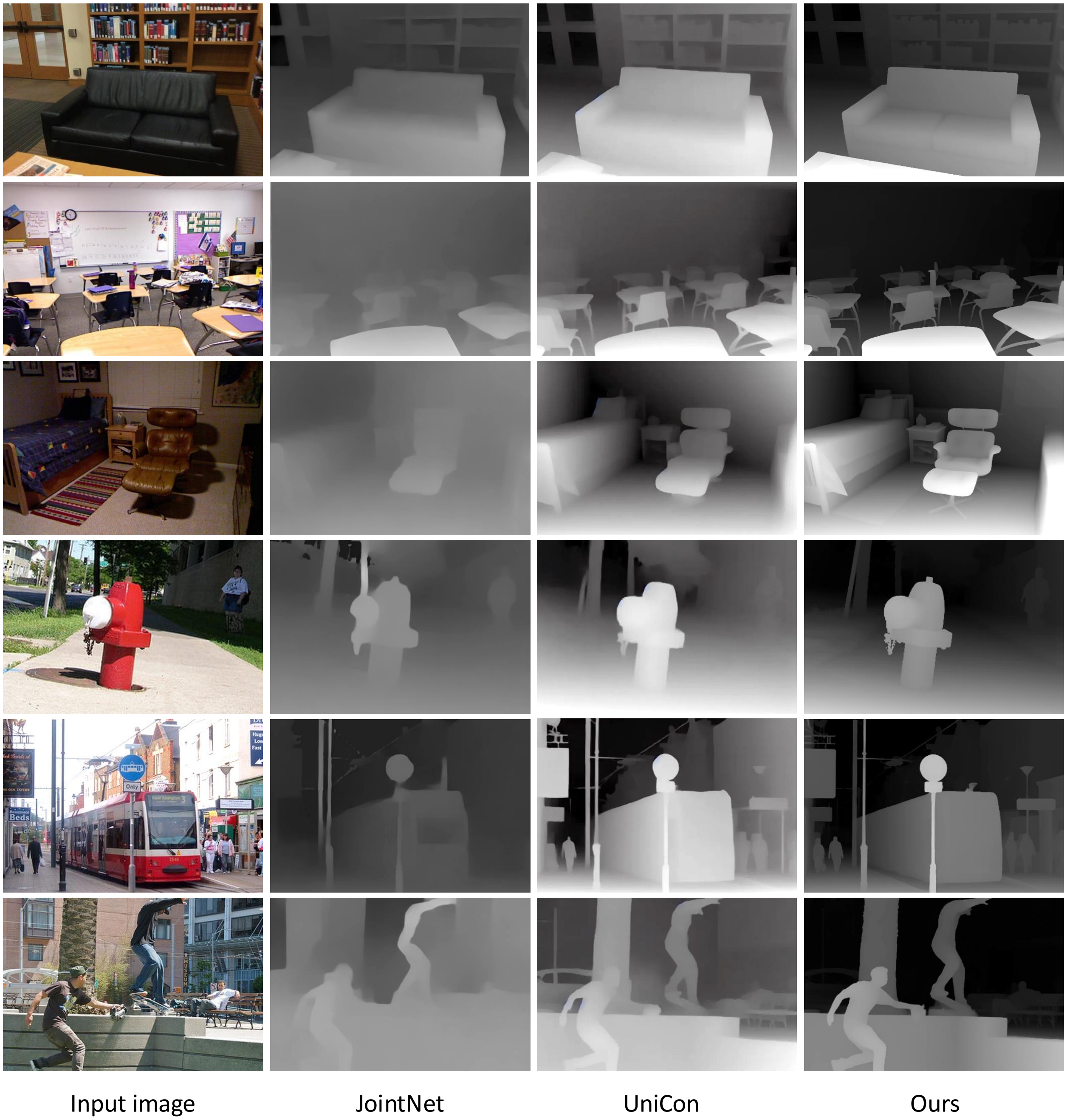

Figure S7. **Depth estimation results of joint generation models.** We visualize the depth estimation results of JointNet, UniCon, and our method on the NYUv2, ScanNet, MSCOCO dataset [4, 18, 28]. Our method shows sharp and fine-detailed depth visualization, which aligns with the trends observed in the qualitative results.

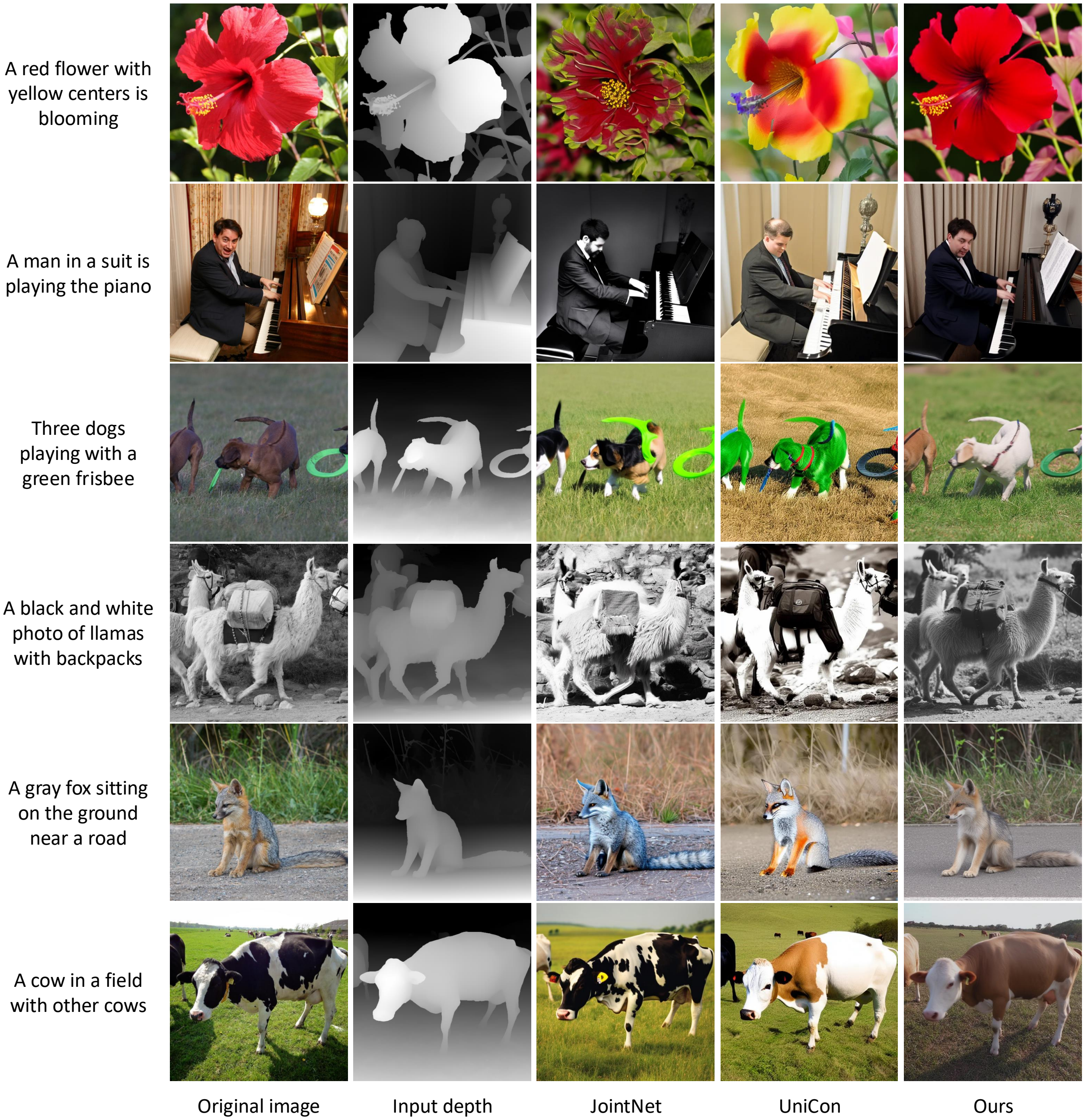
A red flower with yellow centers is blooming
A man in a suit is playing the piano
Three dogs playing with a green frisbee
A black and white photo of llamas with backpacks
A gray fox sitting on the ground near a road
A cow in a field with other cows
Original image
Input depth
JointNet
UniCon
Ours

Figure S8. **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.

References

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

[2] 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. 1

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

[4] 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. 1, 4, 5, 9

[5] 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. 2

[6] 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. 2

[7] 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. 2

[8] Xiao Fu, Wei Yin, Mu Hu, Kaixuan Wang, Yuexin Ma, Ping Tan, Shaojie Shen, Dahua Lin, and Xiaoxiao Long. Geowizard: Unleashing the diffusion priors for 3d geometry estimation from a single image. In *European Conference on Computer Vision*, pages 241–258. Springer, 2024. 1

[9] 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. 3

[10] 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. 2

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

[12] 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. 2

[13] 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. 1, 2, 4

[14] 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. 2

[15] 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. 2

[16] 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. 1, 2, 4

[17] 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. 2

[18] 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. 5, 9

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

[20] Grace Luo, Trevor Darrell, Oliver Wang, Dan B Goldman, and Aleksander Holynski. Readout guidance: Learning control from diffusion features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8217–8227, 2024. 1

[21] 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. 3

[22] 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

[23] 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. 2

[24] 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. 3

[25] 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. 3

[26] 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. 3

[27] 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. 1

[28] 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. 1, 4, 5, 9

[29] 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. 1, 2, 3, 4

[30] 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. 2

[31] 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. 1

[32] 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. 2

[33] 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. 3, 4

[34] 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. 1, 2, 4, 5

[35] 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. 1, 2, 3, 4

[36] 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. 1