

Figure 1. **The proposed DMC-120k dataset.** We first generate the multi-element image with random sampled categories and scenarios and inpaint the elements in order. The conditions are then obtained by corresponding condition extractors.

A. The DMC-120k dataset

The total pipeline of dataset construction is shown in Fig. 1. **Image Creation.** The first step is to generate complex images composed of multiple distinct elements. To achieve this, we define more than 100 different category labels, including animals, furniture, household items, and *etc.*. We then construct prompts based on randomly selected categories and scenarios. These prompts are further optimized using ChatGPT to improve their suitability for the generation model and to enhance the variety of generated prompts. For each prompt, we use a random seed and generate high-resolution images at 1024×1024 pixels, employing the open-source SDXL [1] and FLUX [2] models to generate high-quality image data under various conditions.

Element Decoupling. Then, we use GroundingDino [3] to detect the location and shape of each object in the image. After obtaining the mask, we crop the foreground object and upsample it to 1024×1024 pixels, which is then used to extract the foreground conditions. If two or more foreground element masks overlap, we label them as potentially occluding each other. To address occlusion, we first erase one of the masks and apply SDXL-Inpainting [1] to inpaint the image. We then redetect the remaining objects to ensure they are intact. This process is repeated for all occluding elements until no occlusion remains. To extract the background conditions, we apply SDXL-Inpainting using the masks of all the foreground objects and inpaint the image.

Condition generation. The final step is to obtain the con-

Module	Parameter
SDXL Unet	2.67B
ControlNet	1.34B
Intra-Element Controller	255.50M
Inter-Element Controller	200.20M

Table 1. The parameters in our DC-ControlNet.

ditions. We achieve this by using various condition detectors for both foreground and background images. In DMC-120k, we provide detailed conditions such as canny, HED, depth, segmentation, and normal maps for content control. In addition, we include dot maps, box maps, and mask maps as layout control conditions.

B. Model Size

The trainable part of our model is the Intra-Element Controller and Inter-Element Controller. We present the total parameters on Tab. 1. Note that the roles of different modules are different, so the whole training process is divided into two stages, each focusing on different modules, with unrelated modules frozen in each stage. As two key contributions in the proposed DC-ControlNet, the Intra-Element Controller and Inter-Element Controller contain only about a fifth of the parameters of ControlNet, yet present excellent performance in decoupling controllable image generation.

C. More Visual Results

C.1. The result with different layout

Given different layouts of the element, our DC-ControlNet effectively adjusts the size and position of the foreground. As illustrated in Fig. 2, DC-ControlNet dynamically modifies the relationships between foreground and background elements, such as contact and occlusion, while refining the depth of field. This ensures that the main element stands out and the overall image remains harmonious. The prompt we use in Fig. 2 is “A teddy bear in a cozy living room”.

C.2. Controllable Generation using text

Fig. 3 provides more examples through DC-ControlNet. DC-ControlNet offers great content and layout control for the given elements. Additionally, it is worth noting that DC-ControlNet does not lose the ability of the original diffusion models to control the overall style through text prompt. For example, the user can give the prompt “Anime artwork, anime style, key visual” to create an image in anime style while keeping the elements generated with the given conditions. This demonstrates the strong scalability of our model for user-specific style customization.

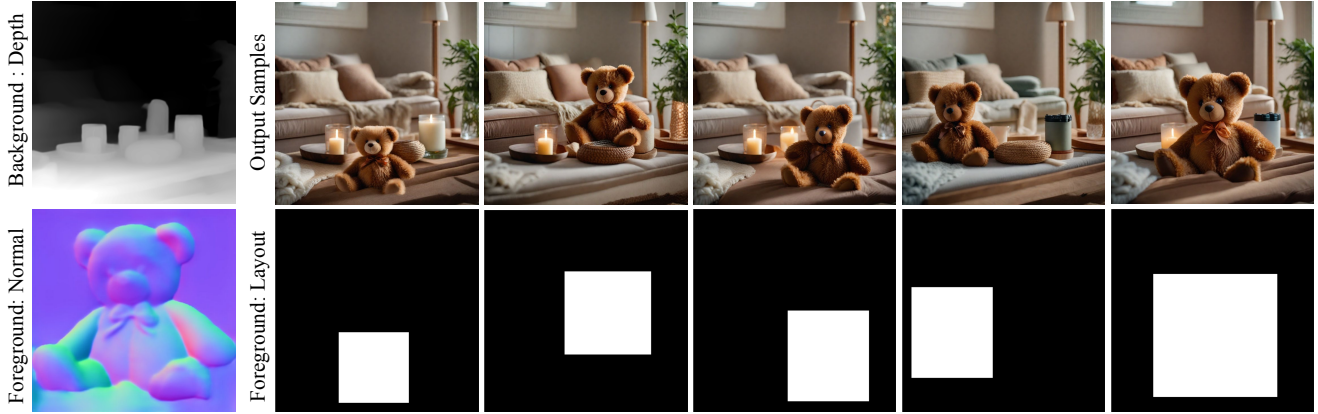


Figure 2. The same element in different layout using our DC-ControlNet. The prompt we use is “A teddy bear in a cozy living room”.

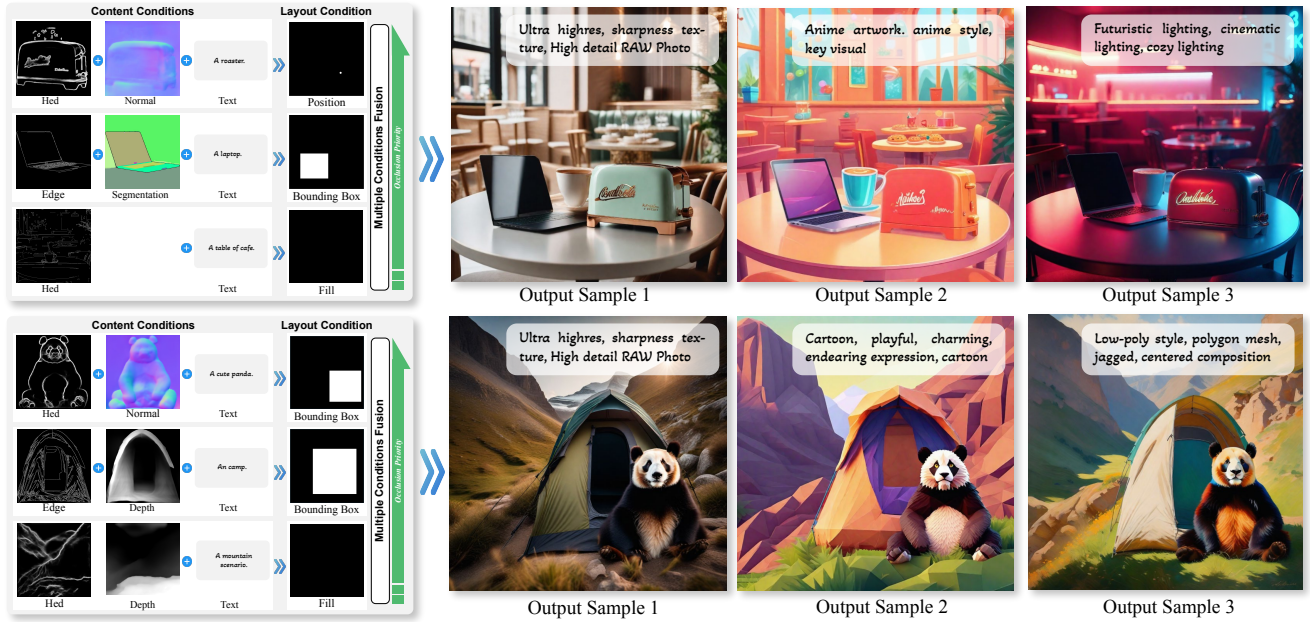


Figure 3. More controllable generation results of our DC-ControlNet. Each example includes an additional style description to generate images with different styles.

C.3. Comparison with existing models

Fig. 4 shows the results with more generative models under the same conditions. ControlNet-based models show significant artifacts when handling multiple conditions and multiple elements, while LayoutDiffusion-based models such as InstanceDiff [10] and GeoDiffusion [11] often produce unnatural and low-quality images. Additionally, LayoutDiffusion models lack control over specific content, leading to misinterpretations of the model. Most importantly, all approaches fail to handle the generation of order relation-

ships. In contrast, our method not only ensures correct relationships, but also produces images with higher quality, more accurate, and richer details, as shown in the red box in the first row of Fig. 4.

Fig. 5 shows another comparison of DC-ControlNet with other existing controllable image generative models. DC-ControlNet significantly outperforms other ControlNet or Layout-to-Image diffusion models in multi-condition, multi-element image generation. Compared to the most competitive models, UniControlNet [6] and HiCo [9], our method shows significant advantages in image harmony and

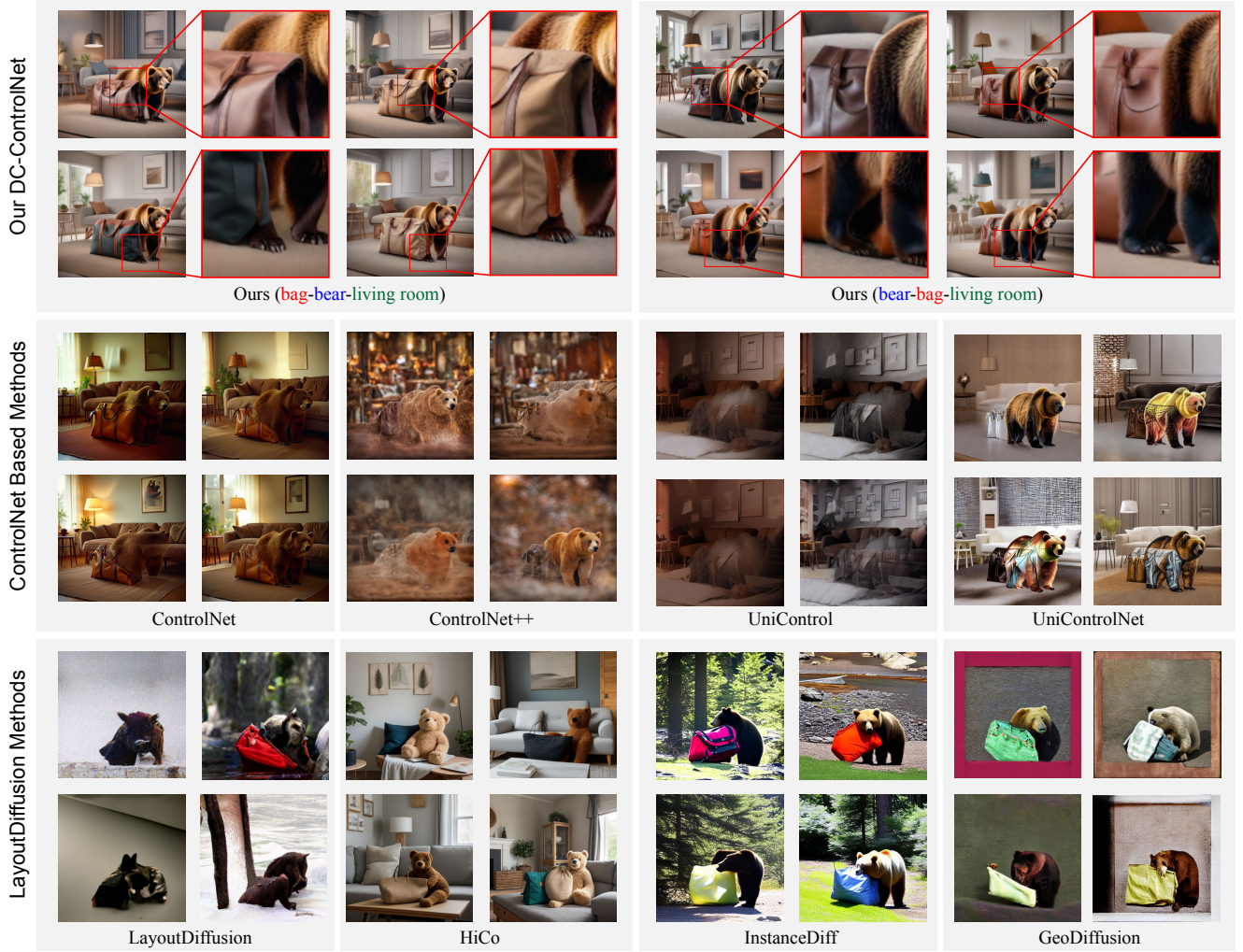


Figure 4. The comparison with ControlNet [4], ControlNet++ [5], UniControl [6], UniControlNet [7], LayoutDiffusion [8], HiCo [9], InstanceDiff [10], and GeoDiffusion [11]. The prompt we used is “A bag and a bear in a cozy living room”. DC-ControlNet ensures correct relationships and produces high quality, accurate, and rich details images.

condition consistency.

C.4. Intra-Element Controller

Fig. 6 illustrates the result of the Intra-Element Controller, our Intra-Element Controller is capable of controlling the outputs of different conditions based on varying layout conditions and generating the element in the desired layout. As a result, controllable image generation is decoupled into content and layout controllable generation, giving users greater flexibility in controlling.

The spatial weights serve to emphasize the main content of each element, while the layer weights highlight which element plays a significant role for each pixel. As the layer relationships between different elements of the same pixel are mutually exclusive, we apply softmax to obtain the final weight, which distinguishes our method from the Spatial

Reweighting Transformer. When obtaining spatial weights, we use a zero-initialized linear layer followed by a sigmoid activation. With these improvements, we achieve precise control by reweighing the features along with both the spatial and layer dimensions. Furthermore, the reweighing approach prevents the model from taking shortcuts by directly modifying the content of specific elements.

D. Ablation Studies

D.1. Intra-Element Controller

The Intra-Element Controller injects the content feature into the corresponding layout through the Cross-Attention mechanism. This process is the conversion of a pixel-misaligned task to a pixel-aligned task at the feature level. The key design of the Intra-Element Controller primarily

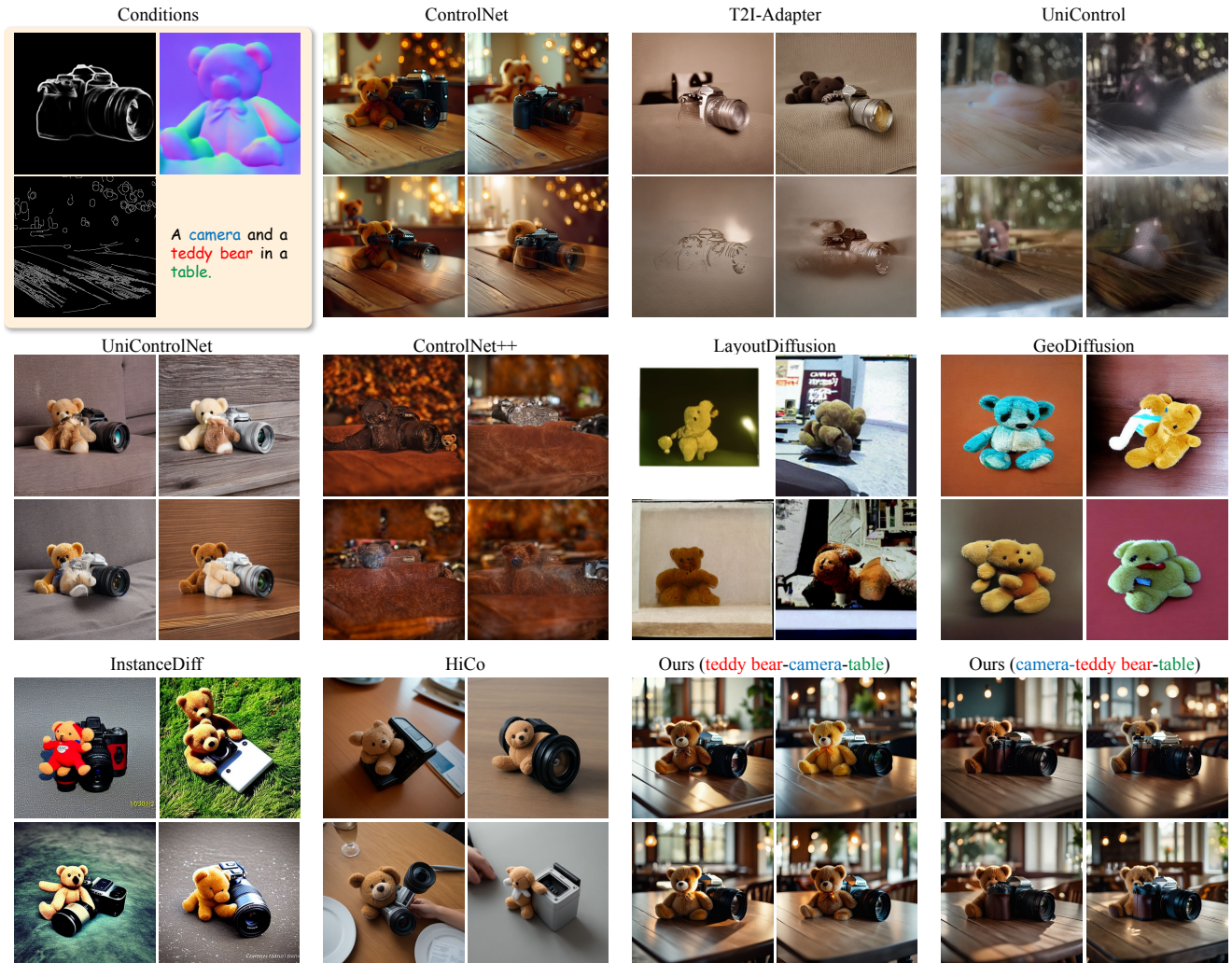


Figure 5. The comparison with ControlNet [4], T2I-Adapter [12], UniControl [6], UniControlNet [7], ControlNet++ [5], LayoutDiffusion [8], InstanceDiff [10], GeoDiffusion [11] and HiCo [9]. The prompt we used is “A camera and a teddy bear in a table”.

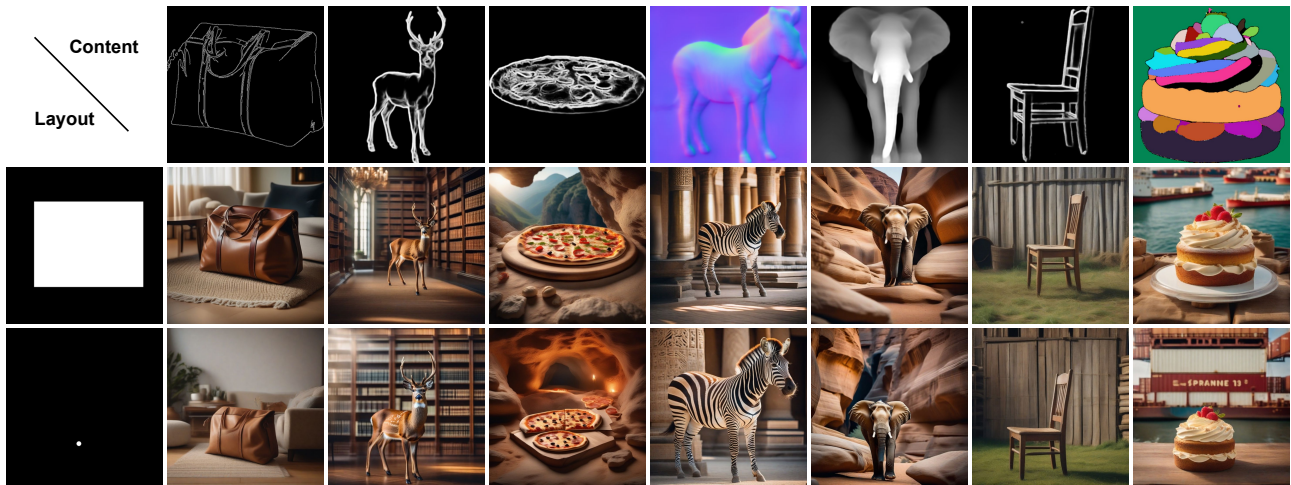


Figure 6. The result of only our Intra-Element Controller. The Intra-Element Controller can effectively and accurately transform the given condition content to the target layout.

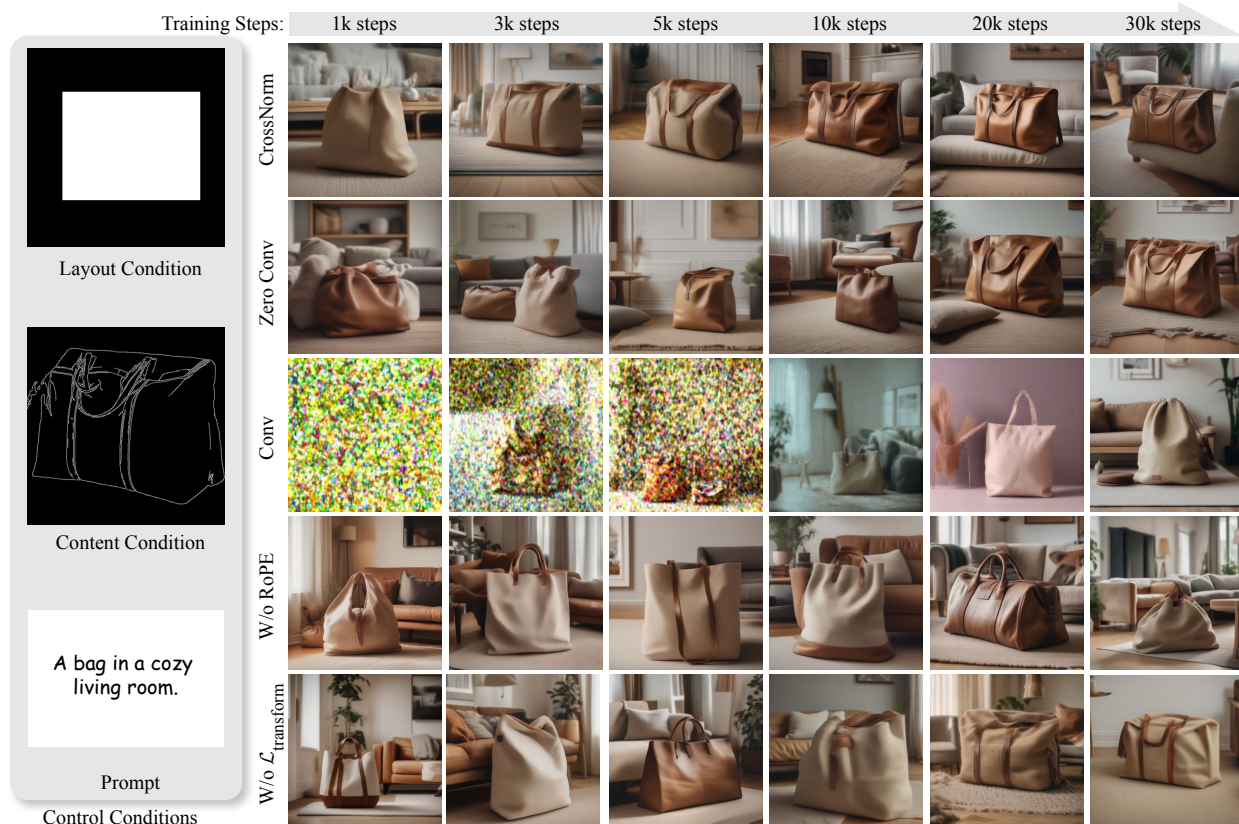


Figure 7. **The ablation study of the Intra-Element Controller.** CrossNorm after the convolution output layer achieves faster convergence compared to that using zero conv or regular conv.

focuses on enabling faster convergence and facilitating the injection of information for misaligned tasks. We present the results in different training steps in Fig. 7. Using CrossNorm allows the model to outline the target structure within 3k training steps and achieve convergence in texture and structure by around 5k steps. This enables the model to focus on the details in the remaining steps. In contrast, models using zero conv require about 20k steps to converge, while variants with standard conv take significantly longer to align the output distribution and half of the training steps are required to produce a reasonable image.

Moreover, not utilizing asynchronous RoPE leads to sub-optimal performance of the Cross-Attention mechanism in misaligned tasks, particularly when the query and key inputs share the same shape. As shown in the fourth row of Fig. 7, the model fails to generate the target in the corresponding layout, and the details of the condition are also missing. Besides, the feature-level transform loss facilitates the model to transfer conditions to the target layout, which can be observed from the fifth row of Fig. 7.

D.2. Inter-Element Controller

The Inter-Element Controller solves two following problems: unnaturalness and occlusion when fusing multiple

elements. To represent the layer ordering relationship between different elements, we assign a 1d order embedding to the sorted element feature, enabling the model to perceive the order relationships between these elements. As shown in Fig. 8, without 1d order embedding, the model misinterprets the order of the elements, leading to a blending problem similar to traditional ControlNet-based models across different elements. The same problem also arises in the absence of the Layer Reweighting transformer, the model cannot distinguish which element should appear, thereby failing to execute the user’s command of positioning a specific element in the foreground. Additionally, the image quality also degrades due to the direct mixing of multiple elements. Besides, the absence of the Spatial Reweighting Transformer may lead to artifacts or unnatural regions in the generated image, which can also be observed in Fig. 8.



Figure 8. **The ablation study of Inter-Element Controller.** Order embedding provides the order information between elements, enabling the model to generate accurate outputs. The absence of the layer transformer and spatial transformer introduces extra artifacts.

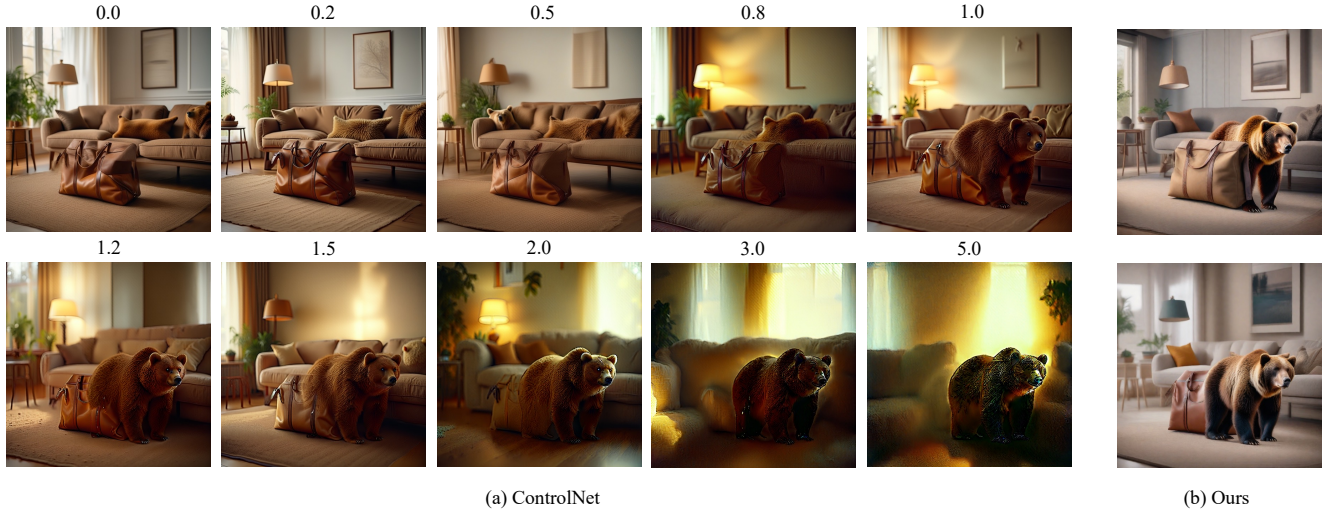


Figure 9. **The comparison between the ControlNet and DC-ControlNet in fusing multiple elements.** The ControlNet uses different condition scales to enhance the element “bear” on the image to place it in front of the “bag” but introduce unexpected artifacts.

E. Discussion

Why is the layer-order way superior to the scale-way? The original ControlNet [4] enhances or weakens the features by simply adjusting their scale. This method, when applied to multiple elements, inevitably leads to unnatural artifacts or distortions. For example, in Fig. 9, with the scale of the feature of “a bear” increase, the ControlNet ultimately achieves the goal of placing a “bear” in front of the “bag”, at the expense of degraded image quality and artifacts. In contrast, our approach applies 1d order embedding to indicate the accurate order relationship of elements, achieving this goal more elegantly and straightforwardly.



Figure 10. **The analysis of the softmax map in the Layer Reweighting Transformer.** The softmax maps are extracted at 64×64 and 32×32 in our Layer Reweighting transformer.

This is because the Intra-Element Controller integrates the weights of various elements and then reweighs these elements. The way in which MultiControlNet [4] fuses multiple elements can be expressed by the following equation, which applies the single scales to weight all conditions.

$$x_{\text{total}} = x_{\text{Unet}} + \sum_{i=0}^L \alpha_i \cdot x_i \quad (1)$$

where x_i and x_{Unet} denote the feature of the i -th element and the main branch feature in the Unet, respectively. And α_i denotes the scale of the i -th element.

In contrast, the Inter-Element Controller incorporates additional Layer/Spatial Reweighting Transformer modules to weight based on all the elements. Its fusing process can be expressed as follows.

$$x_{\text{total}} = x_{\text{Unet}} + \sum_{i=0}^L w_{\text{layer}_i} \cdot (w_{\text{spatial}_i} \cdot x_i) \quad (2)$$

where the w_{spatial_i} and w_{layer_i} denote the weight of spatial and layer dimension, respectively.

How does Layer Reweighting Transformer work? The Layer Reweighting Transformer plays a crucial role in our DC-ControlNet by ensuring that different conditions are processed and fused effectively, particularly when dealing with tasks involving overlapping regions and occlusions. In particular, we apply the softmax function to handle these features in competition. To better illustrate the effectiveness of the fusion, we extract the softmax maps at each layer of the Layer Reweighting Transformer and apply the argmax to visualize which element dominates. As shown in Fig. 10, the weight map obtained via softmax effectively determines which element dominates in order at different pixels, especially in overlapping regions.

F. Graphical User Interface

Fig. 11 shows the graphical user interface in our DC-ControlNet. Since DC-ControlNet offers a more user-friendly and flexible solution for controllable image generations, users can easily add an element and define the properties (such as content, color, and layout). For more details, please refer to our demo video in the Supplementary.

G. Metrics

We use the proposed DMC-120k test set for quantitative evaluations of our design choice. Given that our method is a decoupled ControlNet, we consider the following aspects as our metrics: the accuracy of the generated elements and the order relationships between elements. Therefore, we use the Fréchet Inception Distance (FID) [13], CLIP Score [14], and CLIP Aes [15] metrics to evaluate the quality and accuracy. FID measures the distance between the feature distributions of real and generated images, with lower values indicating higher image quality. CLIP Score evaluates the similarity between the image and text, which is used to evaluate the consistency of the image and the text. CLIP Aes are employed to assess the aesthetics of the images. In particular, we incorporate Large Vision-Language Models (LVLM) as one of the evaluators due to its powerful image-understanding capabilities. ChatGPT [16] is employed to check whether the corresponding elements have been successfully generated and whether the order relationships between elements are correct. For Accuracy_{exist}, we ask the LVLM to determine whether the object is present in the image, the final accuracy is calculated by dividing the number of detected objects by the total number of objects. For Accuracy_{occ}, we require the LVLM to determine which object occludes others, and the final accuracy is calculated by dividing the number of correctly detected occlusions by the total number of occlusions. The prompt is shown in Fig. 12.

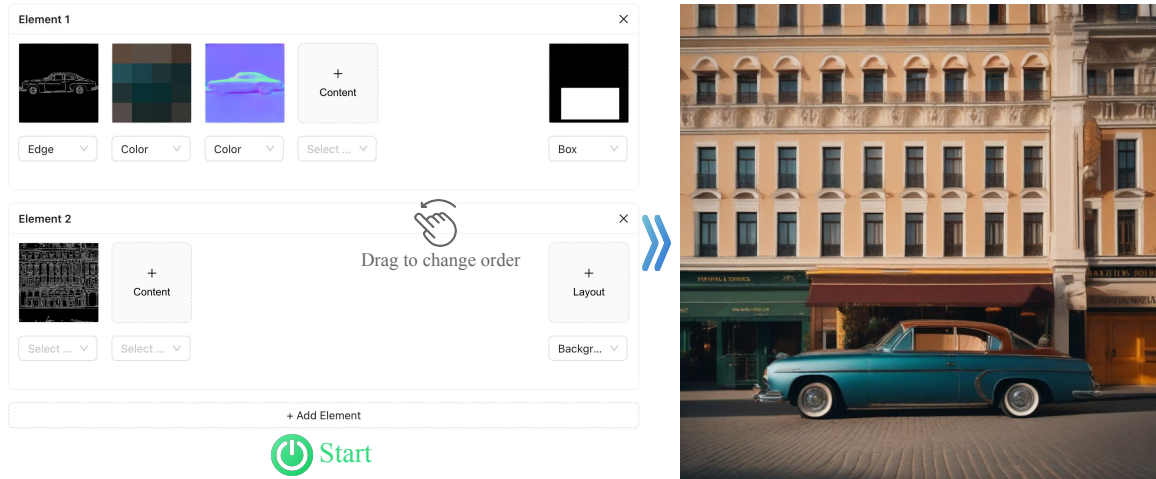


Figure 11. The graphical user interface of our DC-ControlNet. DC-ControlNet offers a user-friendly and flexible solution.

Prompt

You are a helpful and precise assistant for checking the content of the image.

In this image, object A is <OBJECT A>, and object B is <OBJECT B>.

Please answer the following questions in the specified format:

1. Describe the content of the image:

- <Brief description of the image>

2. The existence of object A and object B:

- The existence of object A: <True / False>

- The existence of object B: <True / False>

3. Whether object A and object B overlap:

- The overlap between object A and object B: <True / False / N/A>

4. Front-back relationship:

- If both object A and object B exist and overlap, which one is in front of the other?

- The Front-back relationship: <A / B / N/A>

Figure 12. LVLM prompt in our method to measure the accuracy of the object existence and object occlusion.

References

- [1] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *International Conference on Learning Representations*, 2024. 1
- [2] Black Forest Labs. Flux: Official inference repository for flux.1 models, 2024. Accessed: 2024-11-12. 1
- [3] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55. Springer, 2025. 1
- [4] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *International Conference on Computer Vision*, pages 3836–3847, 2023. 3, 4, 6, 7
- [5] Ming Li, Taojiannan Yang, Huafeng Kuang, Jie Wu, Zhaoning Wang, Xuefeng Xiao, and Chen Chen. Controlnet ++: Improving conditional controls with efficient consistency feedback. In *European Conference on Computer Vision*, pages 129–147. Springer, 2025. 3, 4
- [6] Can Qin, Shu Zhang, Ning Yu, Yihao Feng, Xinyi Yang, Yingbo Zhou, Huan Wang, Juan Carlos Niebles, Caiming Xiong, Silvio Savarese, et al. Unicontrol: a unified diffusion model for controllable visual generation in the wild. In *Neural Information Processing Systems*, pages 42961–42992, 2023. 2, 3, 4
- [7] Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K. Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models. *Neural Information Processing Systems*, 2023. 3, 4
- [8] Guangcong Zheng, Xianpan Zhou, Xuewei Li, Zhongang Qi, Ying Shan, and Xi Li. Layoutdiffusion: Controllable diffusion model for layout-to-image generation. In *Computer Vision and Pattern Recognition*, pages 22490–22499, 2023. 3, 4
- [9] Bo Cheng, Yuhang Ma, Liebuha Wu, Shanyuan Liu, Ao Ma, Xiaoyu Wu, Dawei Leng, and Yuhui Yin. Hico: Hierarchical controllable diffusion model for layout-to-image generation. *arXiv preprint arXiv:2410.14324*, 2024. 2, 3, 4
- [10] Xudong Wang, Trevor Darrell, Sai Saketh Rambhatla, Rohit Girdhar, and Ishan Misra. Instancediffusion: Instance-level control for image generation. In *Computer Vision and Pattern Recognition*, pages 6232–6242, 2024. 2, 3, 4
- [11] Kai Chen, Enze Xie, Zhe Chen, Yibo Wang, Lanqing Hong, Zhenguo Li, and Dit-Yan Yeung. Geodiffusion: Text-

prompted geometric control for object detection data generation. In *International Conference on Learning Representations*, 2024. [2](#), [3](#), [4](#)

- [12] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *the AAAI Conference on Artificial Intelligence*, pages 4296–4304, 2024. [4](#)
- [13] 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. *Neural Information Processing Systems*, 30, 2017. [7](#)
- [14] 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, 2021. [7](#)
- [15] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Neural Information Processing Systems*, 35:25278–25294, 2022. [7](#)
- [16] OpenAI. Chatgpt, 2024. Accessed: 2024-07-12. [7](#)