Adding Conditional Control to Text-to-Image Diffusion Models

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala
Stanford University
{lvmin, anyirao, maneesh}@cs.stanford.edu

Figure 1: Controlling Stable Diffusion with learned conditions. ControlNet allows users to add conditions like Canny edges (top), human pose (bottom), *etc*., to control the image generation of large pretrained diffusion models. The default results use the prompt "a high-quality, detailed, and professional image". Users can optionally give prompts like the "chef in kitchen".

Abstract

We present ControlNet, a neural network architecture to add spatial conditioning controls to large, pretrained text-to-image diffusion models. ControlNet locks the production-ready large diffusion models, and reuses their deep and robust encoding layers pretrained with billions of images as a strong backbone to learn a diverse set of conditional controls. The neural architecture is connected with "zero convolutions" (zero-initialized convolution layers) that progressively grow the parameters from zero and ensure that no harmful noise could affect the finetuning. We test various conditioning controls, e.g.*, edges, depth, segmentation, human pose,* etc.*, with Stable Diffusion, using single or multiple conditions, with or without prompts. We show that the training of ControlNets is robust with small (<50k) and large (>1m) datasets. Extensive results show that ControlNet may facilitate wider applications to control image diffusion models.*

1. Introduction

Many of us have experienced flashes of visual inspiration that we wish to capture in a unique image. With the advent of text-to-image diffusion models [54, 62, 72], we can now create visually stunning images by typing in a text prompt. Yet, text-to-image models are limited in the control they provide over the spatial composition of the image; precisely expressing complex layouts, poses, shapes and forms can be difficult via text prompts alone. Generating an image that accurately matches our mental imagery often requires numerous trial-and-error cycles of editing a prompt, inspecting the resulting images and then re-editing the prompt.

Can we enable finer grained spatial control by letting users provide additional images that directly specify their desired image composition? In computer vision and machine learning, these additional images (*e.g*., edge maps, human pose skeletons, segmentation maps, depth, normals, *etc*.) are often treated as conditioning on the image generation process. Image-to-image translation models [34, 98] learn

the mapping from conditioning images to target images. The research community has also taken steps to control text-to-image models with spatial masks [6, 20], image editing instructions [10], personalization via finetuning [21, 75], *etc*. While a few problems (*e.g*., generating image variations, inpainting) can be resolved with training-free techniques like constraining the denoising diffusion process or editing attention layer activations, a wider variety of problems like depth-to-image, pose-to-image, *etc*., require end-to-end learning and data-driven solutions.

Learning conditional controls for large text-to-image diffusion models in an end-to-end way is challenging. The amount of training data for a specific condition may be significantly smaller than the data available for general text-to-image training. For instance, the largest datasets for various specific problems (*e.g*., object shape/normal, human pose extraction, *etc*.) are usually about 100K in size, which is 50,000 times smaller than the LAION-5B [79] dataset that was used to train Stable Diffusion [82]. The direct finetuning or continued training of a large pretrained model with limited data may cause overfitting and catastrophic forgetting [31, 75]. Researchers have shown that such forgetting can be alleviated by restricting the number or rank of trainable parameters [14, 25, 31, 92]. For our problem, designing deeper or more customized neural architectures might be necessary for handling in-the-wild conditioning images with complex shapes and diverse high-level semantics.

This paper presents ControlNet, an end-to-end neural network architecture that learns conditional controls for large pretrained text-to-image diffusion models (Stable Diffusion in our implementation). ControlNet preserves the quality and capabilities of the large model by locking its parameters, and also making a *trainable copy* of its encoding layers. This architecture treats the large pretrained model as a strong backbone for learning diverse conditional controls. The trainable copy and the original, locked model are connected with *zero convolution* layers, with weights initialized to zeros so that they progressively grow during the training. This architecture ensures that harmful noise is not added to the deep features of the large diffusion model at the beginning of training, and protects the large-scale pretrained backbone in the trainable copy from being damaged by such noise.

Our experiments show that ControlNet can control Stable Diffusion with various conditioning inputs, including Canny edges, Hough lines, user scribbles, human key points, segmentation maps, shape normals, depths, *etc*. (Figure 1). We test our approach using a single conditioning image, with or without text prompts, and we demonstrate how our approach supports the composition of multiple conditions. Additionally, we report that the training of ControlNet is robust and scalable on datasets of different sizes, and that for some tasks like depth-to-image conditioning, training ControlNets on a single NVIDIA RTX 3090Ti GPU can achieve results competitive with industrial models trained on large computation clusters. Finally, we conduct ablative studies to investigate the contribution of each component of our model, and compare our models to several strong conditional image generation baselines with user studies.

In summary, (1) we propose ControlNet, a neural network architecture that can add spatially localized input conditions to a pretrained text-to-image diffusion model via efficient finetuning, (2) we present pretrained ControlNets to control Stable Diffusion, conditioned on Canny edges, Hough lines, user scribbles, human key points, segmentation maps, shape normals, depths, and cartoon line drawings, and (3) we validate the method with ablative experiments comparing to several alternative architectures, and conduct user studies focused on several previous baselines across different tasks.

2. Related Work

2.1. Finetuning Neural Networks

One way to finetune a neural network is to directly continue training it with the additional training data. But this approach can lead to overfitting, mode collapse, and catastrophic forgetting. Extensive research has focused on developing finetuning strategies that avoid such issues.

HyperNetwork is an approach that originated in the Natural Language Processing (NLP) community [25], with the aim of training a small recurrent neural network to influence the weights of a larger one. It has been applied to image generation with generative adversarial networks (GANs) [4, 18]. Heathen *et al*. [26] and Kurumuz [43] implement HyperNetworks for Stable Diffusion [72] to change the artistic style of its output images.

Adapter methods are widely used in NLP for customizing a pretrained transformer model to other tasks by embedding new module layers into it [30, 84]. In computer vision, adapters are used for incremental learning [74] and domain adaptation [70]. This technique is often used with CLIP [66] for transferring pretrained backbone models to different tasks [23, 66, 85, 94]. More recently, adapters have yielded successful results in vision transformers [49, 50] and ViT-Adapter [14]. In concurrent work with ours, T2I-Adapter [56] adapts Stable Diffusion to external conditions.

Additive Learning circumvents forgetting by freezing the original model weights and adding a small number of new parameters using learned weight masks [51, 74], pruning [52], or hard attention [80]. Side-Tuning [92] uses a side branch model to learn extra functionality by linearly blending the outputs of a frozen model and an added network, with a predefined blending weight schedule.

Low-Rank Adaptation (LoRA) prevents catastrophic forgetting [31] by learning the offset of parameters with low-rank matrices, based on the observation that many over-

parameterized models reside in a low intrinsic dimension subspace [2, 47].

Zero-Initialized Layers are used by ControlNet for connecting network blocks. Research on neural networks has extensively discussed the initialization and manipulation of network weights [36, 37, 44, 45, 46, 76, 83, 95]. For example, Gaussian initialization of weights can be less risky than initializing with zeros [1]. More recently, Nichol *et al*. [59] discussed how to scale the initial weight of convolution layers in a diffusion model to improve the training, and their implementation of "zero_module" is an extreme case to scale weights to zero. Stability's model cards [83] also mention the use of zero weights in neural layers. Manipulating the initial convolution weights is also discussed in ProGAN [36], StyleGAN [37], and Noise2Noise [46].

2.2. Image Diffusion

Image Diffusion Models were first introduced by Sohl-Dickstein *et al*. [81] and have been recently applied to image generation [17, 42]. The Latent Diffusion Models (LDM) [72] performs the diffusion steps in the latent image space [19], which reduces the computation cost. Text-to-image diffusion models achieve state-of-the-art image generation results by encoding text inputs into latent vectors via pretrained language models like CLIP [66]. Glide [58] is a text-guided diffusion model supporting image generation and editing. Disco Diffusion [5] processes text prompts with clip guidance. Stable Diffusion [82] is a large-scale implementation of latent diffusion [72]. Imagen [78] directly diffuses pixels using a pyramid structure without using latent images. Commercial products include DALL-E2 [62] and Midjourney [54].

Controlling Image Diffusion Models facilitate personalization, customization, or task-specific image generation. The image diffusion process directly provides some control over color variation [53] and inpainting [67, 7]. Text-guided control methods focus on adjusting prompts, manipulating CLIP features, and modifying cross-attention [7, 10, 20, 27, 40, 41, 58, 64, 67]. MakeAScene [20] encodes segmentation masks into tokens to control image generation. SpaText [6] maps segmentation masks into localized token embeddings. GLIGEN [48] learns new parameters in attention layers of diffusion models for grounded generating. Textual Inversion [21] and DreamBooth [75] can personalize content in the generated image by finetuning the image diffusion model using a small set of user-provided example images. Prompt-based image editing [10, 33, 86] provides practical tools to manipulate images with prompts. Voynov *et al*. [88] propose an optimization method that fits the diffusion process with sketches. Concurrent works [8, 9, 32, 56] examine a wide variety of ways to control diffusion models.

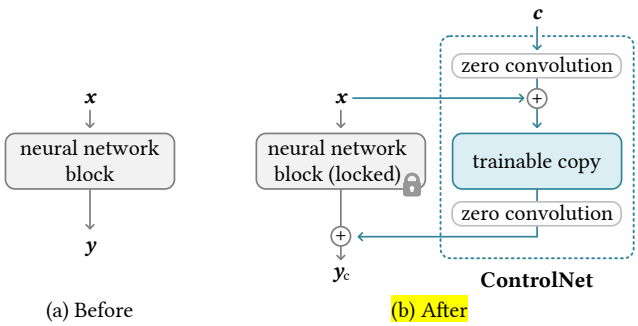

Figure 2: A neural block takes a feature map x as input and outputs another feature map y, as shown in (a). To add a ControlNet to such a block we lock the original block and create a trainable copy and connect them together using zero convolution layers, *i.e*., 1×1 convolution with both weight and bias initialized to zero. Here c is a conditioning vector that we wish to add to the network, as shown in (b).

2.3. Image-to-Image Translation

Conditional GANs [15, 34, 63, 90, 93, 97, 98, 99] and transformers [13, 19, 68] can learn the mapping between different image domains, *e.g*., Taming Transformer [19] is a vision transformer approach; Palette [77] is a conditional diffusion model trained from scratch; PITI [89] is a pretraining-based conditional diffusion model for image-to-image translation. Manipulating pretrained GANs can handle specific image-to-image tasks, *e.g*., StyleGANs can be controlled by extra encoders [71], with more applications studied in [3, 22, 38, 39, 55, 60, 65, 71].

3. Method

ControlNet is a neural network architecture that can enhance large pretrained text-to-image diffusion models with spatially localized, task-specific image conditions. We first introduce the basic structure of a ControlNet in Section 3.1 and then describe how we apply a ControlNet to the image diffusion model Stable Diffusion [72] in Section 3.2. We elaborate on our training in Section 3.3 and detail several extra considerations during inference such as composing multiple ControlNets in Section 3.4.

3.1. ControlNet

ControlNet injects additional conditions into the blocks of a neural network (Figure 2). Herein, we use the term *network block* to refer to a set of neural layers that are commonly put together to form a single unit of a neural network, *e.g*., resnet block, conv-bn-relu block, multi-head attention block, transformer block, *etc*. Suppose $\mathcal{F}(\cdot;\Theta)$ is such a trained neural block, with parameters Θ, that transforms an input feature map $\boldsymbol{x}$, into another feature map $\boldsymbol{y}$ as

$$\boldsymbol{y} = \mathcal{F}(\boldsymbol{x};\Theta). \tag{1}$$

In our setting, $\boldsymbol{x}$ and $\boldsymbol{y}$ are usually 2D feature maps, *i.e.*, $\boldsymbol{x} \in \mathbb{R}^{h \times w \times c}$ with $\{h, w, c\}$ as the height, width, and number of channels in the map, respectively (Figure 2a).

To add a ControlNet to such a pre-trained neural block, we lock (freeze) the parameters Θ of the original block and simultaneously clone the block to a *trainable copy* with parameters Θ_c (Figure 2b). The trainable copy takes an external conditioning vector $\boldsymbol{c}$ as input. When this structure is applied to large models like Stable Diffusion, the locked parameters preserve the production-ready model trained with billions of images, while the trainable copy reuses such large-scale pretrained model to establish a deep, robust, and strong backbone for handling diverse input conditions.

The trainable copy is connected to the locked model with *zero convolution* layers, denoted $\mathcal{Z}(\cdot;\cdot)$. Specifically, $\mathcal{Z}(\cdot;\cdot)$ is a 1×1 convolution layer with both weight and bias initialized to zeros. To build up a ControlNet, we use two instances of zero convolutions with parameters Θ_{z1} and Θ_{z2} respectively. The complete ControlNet then computes

$$\boldsymbol{y}_c = \mathcal{F}(\boldsymbol{x}; \Theta) + \mathcal{Z}(\mathcal{F}(\boldsymbol{x} + \mathcal{Z}(\boldsymbol{c}; \Theta_{z1}); \Theta_c); \Theta_{z2}), \quad (2)$$

where $\boldsymbol{y}_c$ is the output of the ControlNet block. In the first training step, since both the weight and bias parameters of a zero convolution layer are initialized to zero, both of the $\mathcal{Z}(\cdot;\cdot)$ terms in Equation (2) evaluate to zero, and

$$\boldsymbol{y}_c = \boldsymbol{y}. \quad (3)$$

In this way, harmful noise cannot influence the hidden states of the neural network layers in the trainable copy when the training starts. Moreover, since $\mathcal{Z}(\boldsymbol{c}; \Theta_{z1}) = \boldsymbol{0}$ and the trainable copy also receives the input image $\boldsymbol{x}$, the trainable copy is fully functional and retains the capabilities of the large, pretrained model allowing it to serve as a strong backbone for further learning. Zero convolutions protect this backbone by eliminating random noise as gradients in the initial training steps. We detail the gradient calculation for zero convolutions in supplementary materials.

3.2. ControlNet for Text-to-Image Diffusion

We use Stable Diffusion [72] as an example to show how ControlNet can add conditional control to a large pretrained diffusion model. Stable Diffusion is essentially a U-Net [73] with an encoder, a middle block, and a skip-connected decoder. Both the encoder and decoder contain 12 blocks, and the full model contains 25 blocks, including the middle block. Of the 25 blocks, 8 blocks are down-sampling or up-sampling convolution layers, while the other 17 blocks are main blocks that each contain 4 resnet layers and 2 Vision Transformers (ViTs). Each ViT contains several cross-attention and self-attention mechanisms. For example, in Figure 3a, the "SD Encoder Block A" contains 4 resnet layers and 2 ViTs, while the "×3" indicates that this block is repeated three times. Text prompts are encoded using the

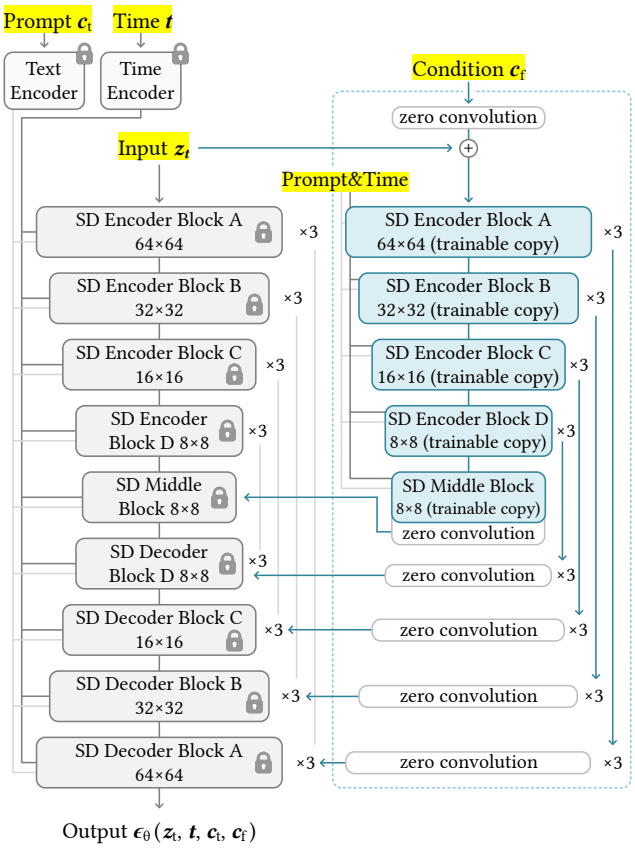

Figure 3: Stable Diffusion's U-net architecture connected with a ControlNet on the encoder blocks and middle block. The locked, gray blocks show the structure of Stable Diffusion V1.5 (or V2.1, as they use the same U-net architecture). The trainable blue blocks and the white zero convolution layers are added to build a ControlNet.

CLIP text encoder [66], and diffusion timesteps are encoded with a time encoder using positional encoding.

The ControlNet structure is applied to each encoder level of the U-net (Figure 3b). In particular, we use ControlNet to create a trainable copy of the 12 encoding blocks and 1 middle block of Stable Diffusion. The 12 encoding blocks are in 4 resolutions ($64 \times 64, 32 \times 32, 16 \times 16, 8 \times 8$) with each one replicated 3 times. The outputs are added to the 12 skip-connections and 1 middle block of the U-net. Since Stable Diffusion is a typical U-net structure, this ControlNet architecture is likely to be applicable with other models.

The way we connect the ControlNet is computationally efficient — since the locked copy parameters are frozen, no gradient computation is required in the originally locked encoder for the finetuning. This approach speeds up training and saves GPU memory. As tested on a single NVIDIA A100 PCIE 40GB, optimizing Stable Diffusion with ControlNet requires only about 23% more GPU memory and 34%

more time in each training iteration, compared to optimizing Stable Diffusion without ControlNet.

Image diffusion models learn to progressively denoise images and generate samples from the training domain. The denoising process can occur in pixel space or in a *latent* space encoded from training data. Stable Diffusion uses latent images as the training domain as working in this space has been shown to stabilize the training process [72]. Specifically, Stable Diffusion uses a pre-processing method similar to VQ-GAN [19] to convert 512×512 pixel-space images into smaller 64×64 *latent images*. To add ControlNet to Stable Diffusion, we first convert each input conditioning image (*e.g.*, edge, pose, depth, *etc.*) from an input size of 512×512 into a 64×64 feature space vector that matches the size of Stable Diffusion. In particular, we use a tiny network $\mathcal{E}(\cdot)$ of four convolution layers with 4×4 kernels and 2×2 strides (activated by ReLU, using 16, 32, 64, 128, channels respectively, initialized with Gaussian weights and trained jointly with the full model) to encode an image-space condition $\boldsymbol{c}_\text{i}$ into a feature space conditioning vector $\boldsymbol{c}_\text{f}$ as,

$$\boldsymbol{c}_\text{f} = \mathcal{E}(\boldsymbol{c}_\text{i}). \tag{4}$$

The conditioning vector $\boldsymbol{c}_\text{f}$ is passed into the ControlNet.

3.3. Training

Given an input image $\boldsymbol{z}_0$, image diffusion algorithms progressively add noise to the image and produce a noisy image $\boldsymbol{z}_t$, where t represents the number of times noise is added. Given a set of conditions including time step $\boldsymbol{t}$, text prompts $\boldsymbol{c}_t$, as well as a task-specific condition $\boldsymbol{c}_\text{f}$, image diffusion algorithms learn a network ϵ_θ to predict the noise added to the noisy image $\boldsymbol{z}_t$ with

$$\mathcal{L} = \mathbb{E}_{\boldsymbol{z}_0, \boldsymbol{t}, \boldsymbol{c}_t, \boldsymbol{c}_\text{f}, \epsilon \sim \mathcal{N}(0,1)} \Big[\| \epsilon - \epsilon_\theta(\boldsymbol{z}_t, \boldsymbol{t}, \boldsymbol{c}_t, \boldsymbol{c}_\text{f})) \|_2^2 \Big], \tag{5}$$

where $\mathcal{L}$ is the overall learning objective of the entire diffusion model. This learning objective is directly used in finetuning diffusion models with ControlNet.

In the training process, we randomly replace 50% text prompts $\boldsymbol{c}_t$ with empty strings. This approach increases ControlNet's ability to directly recognize semantics in the input conditioning images (*e.g.*, edges, poses, depth, *etc.*) as a replacement for the prompt.

During the training process, since zero convolutions do not add noise to the network, the model should always be able to predict high-quality images. We observe that the model does not gradually learn the control conditions but abruptly succeeds in following the input conditioning image; usually in less than 10K optimization steps. As shown in Figure 4, we call this the "sudden convergence phenomenon".

3.4. Inference

We can further control how the extra conditions of ControlNet affect the denoising diffusion process in several ways.

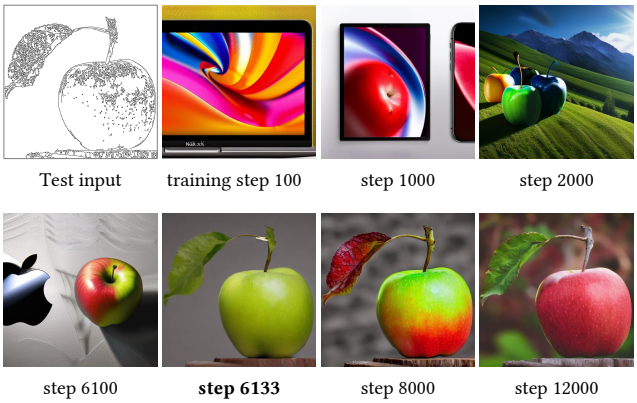

Figure 4: The sudden convergence phenomenon. Due to the zero convolutions, ControlNet always predicts high-quality images during the entire training. At a certain step in the training process (*e.g.*, the 6133 steps marked in bold), the model suddenly learns to follow the input condition.

Figure 5: Effect of Classifier-Free Guidance (CFG) and the proposed CFG Resolution Weighting (CFG-RW).

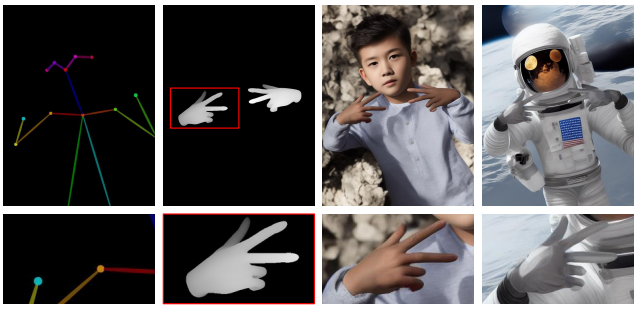

Figure 6: Composition of multiple conditions. We present the application to use depth and pose simultaneously.

Classifier-free guidance resolution weighting. Stable Diffusion depends on a technique called Classifier-Free Guidance (CFG) [29] to generate high-quality images. CFG is formulated as $\epsilon_\text{prd} = \epsilon_\text{uc} + \beta_\text{cfg}(\epsilon_\text{c} - \epsilon_\text{uc})$ where ϵ_prd, ϵ_uc, ϵ_c, β_cfg are the model's final output, unconditional output, conditional output, and a user-specified weight respectively. When a conditioning image is added via ControlNet, it can be added to both ϵ_uc and ϵ_c, or only to the ϵ_c. In challenging cases, *e.g.*, when no prompts are given, adding it to both ϵ_uc and ϵ_c will completely remove CFG guidance (Figure 5b); using only ϵ_c will make the guidance very strong (Figure 5c). Our solution is to first add the conditioning image to ϵ_c and

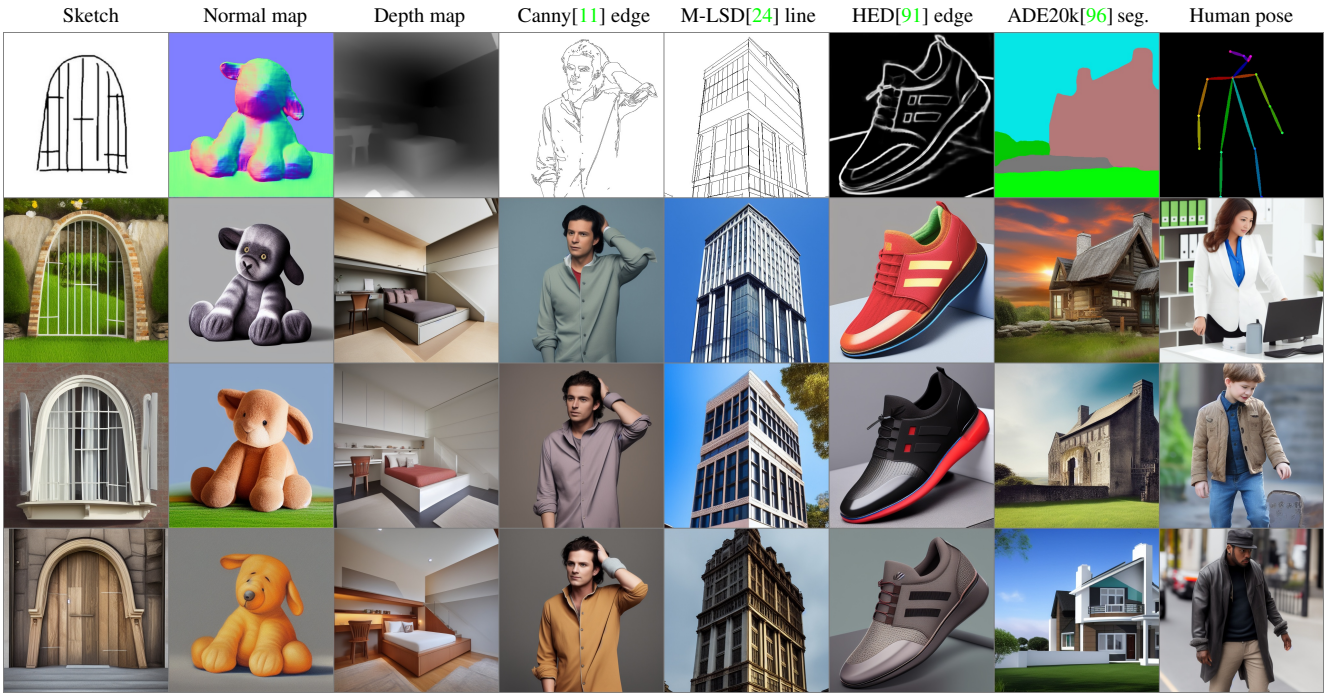

Figure 7: Controlling Stable Diffusion with various conditions **without prompts**. The top row is input conditions, while all other rows are outputs. We use the empty string as input prompts. All models are trained with general-domain data. The model has to recognize semantic contents in the input condition images to generate images.

Method	Result Quality ↑	Condition Fidelity ↑
PITI [89](sketch)	1.10 ± 0.05	1.02 ± 0.01
Sketch-Guided [88] ($\beta = 1.6$)	3.21 ± 0.62	2.31 ± 0.57
Sketch-Guided [88] ($\beta = 3.2$)	2.52 ± 0.44	3.28 ± 0.72
ControlNet-lite	3.93 ± 0.59	4.09 ± 0.46
ControlNet	$\mathbf{4.22 \pm 0.43}$	$\mathbf{4.28 \pm 0.45}$

Table 1: Average User Ranking (AUR) of result quality and condition fidelity. We report the user preference ranking (1 to 5 indicates worst to best) of different methods.

then multiply a weight w_i to each connection between Stable Diffusion and ControlNet according to the resolution of each block $w_i = 64/h_i$, where h_i is the size of i^{th} block, *e.g.*, $h_1 = 8, h_2 = 16, ..., h_{13} = 64$. By reducing the CFG guidance strength , we can achieve the result shown in Figure 5d, and we call this CFG Resolution Weighting.

Composing multiple ControlNets. To apply multiple conditioning images (*e.g.*, Canny edges, and pose) to a single instance of Stable Diffusion, we can directly add the outputs of the corresponding ControlNets to the Stable Diffusion model (Figure 6). No extra weighting or linear interpolation is necessary for such composition.

4. Experiments

We implement ControlNets with Stable Diffusion to test various conditions, including Canny Edge [11], Depth Map [69], Normal Map [87], M-LSD lines [24], HED soft edge [91], ADE20K segmentation [96], Openpose [12], and user sketches. See also the supplementary material for examples of each conditioning along with detailed training and inference parameters.

4.1. Qualitative Results

Figure 1 shows the generated images in several prompt settings. Figure 7 shows our results with various conditions without prompts, where the ControlNet robustly interprets content semantics in diverse input conditioning images.

4.2. Ablative Study

We study alternative structures of ControlNets by (1) replacing the zero convolutions with standard convolution layers initialized with Gaussian weights, and (2) replacing each block's trainable copy with one single convolution layer, which we call ControlNet-lite. See also the supplementary material for the full details of these ablative structures.

We present 4 prompt settings to test with possible behaviors of real-world users: (1) no prompt; (2) insufficient prompts that do not fully cover objects in conditioning images, *e.g.*, the default prompt of this paper "a high-quality, detailed, and professional image"; (3) conflicting prompts that change the semantics of conditioning images; (4) perfect prompts that describe necessary content semantics, *e.g.*, "a nice house". Figure 8a shows that ControlNet succeeds in

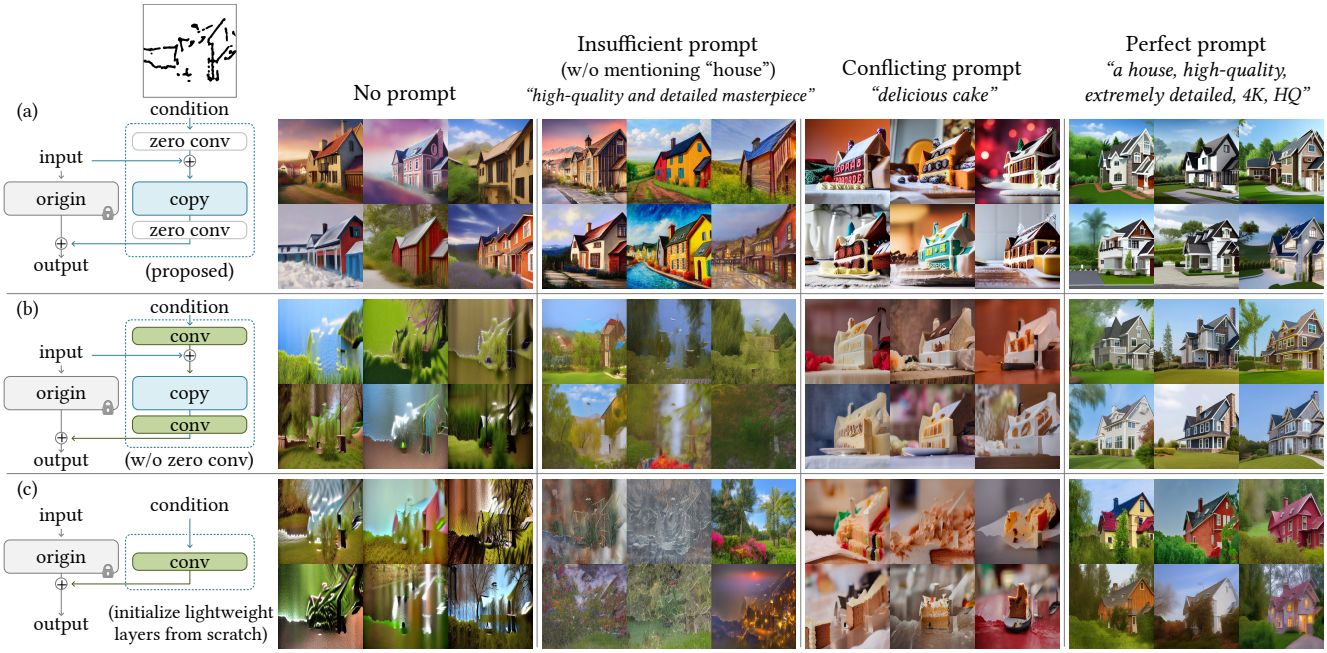

Figure 8: Ablative study of different architectures on a sketch condition and different prompt settings. For each setting, we show a random batch of 6 samples without cherry-picking. Images are at 512 × 512 and best viewed when zoomed in. The green "conv" blocks on the left are standard convolution layers initialized with Gaussian weights.

ADE20K (GT)	VQGAN [19]	LDM [72]	PITI [89]	ControlNet-lite	ControlNet
0.58 ± 0.10	0.21 ± 0.15	0.31 ± 0.09	0.26 ± 0.16	0.32 ± 0.12	**0.35 ± 0.14**

Table 2: Evaluation of semantic segmentation label reconstruction (ADE20K) with Intersection over Union (IoU ↑).

all 4 settings. The lightweight ControlNet-lite (Figure 8c) is not strong enough to interpret the conditioning images and fails in the insufficient and no prompt conditions. When zero convolutions are replaced, the performance of ControlNet drops to about the same as ControlNet-lite, indicating that the pretrained backbone of the trainable copy is destroyed during finetuning (Figure 8b).

4.3. Quantitative Evaluation

User study. We sample 20 unseen hand-drawn sketches, and then assign each sketch to 5 methods: PITI [89]'s sketch model, Sketch-Guided Diffusion (SGD) [88] with default edge-guidance scale ($\beta = 1.6$), SGD [88] with relatively high edge-guidance scale ($\beta = 3.2$), the aforementioned ControlNet-lite, and ControlNet. We invited 12 users to rank these 20 groups of 5 results individually in terms of *"the quality of displayed images"* and *"the fidelity to the sketch"*. In this way, we obtain 100 rankings for result quality and 100 for condition fidelity. We use the Average Human Ranking (AHR) as a preference metric where users rank each result on a scale of 1 to 5 (lower is worse). The average rankings are shown in Table 1.

Comparison to industrial models. Stable Diffusion V2 Depth-to-Image (SDv2-D2I) [83] is trained with a large-scale NVIDIA A100 cluster, thousands of GPU hours, and more than 12M training images. We train a ControlNet for the SD V2 with the same depth conditioning but only use 200k training samples, one single NVIDIA RTX 3090Ti, and 5 days of training. We use 100 images generated by each SDv2-D2I and ControlNet to teach 12 users to distinguish the two methods. Afterwards, we generate 200 images and ask the users to tell which model generated each image. The average precision of the users is 0.52 ± 0.17, indicating that the two method yields almost indistinguishable results.

Method	FID ↓	CLIP-score ↑	CLIP-aes. ↑
Stable Diffusion	6.09	0.26	6.32
VQGAN [19](seg.)*	26.28	0.17	5.14
LDM [72](seg.)*	25.35	0.18	5.15
PITI [89](seg.)	19.74	0.20	5.77
ControlNet-lite	17.92	0.26	6.30
ControlNet	15.27	0.26	6.31

Table 3: Evaluation for image generation conditioned by semantic segmentation. We report FID, CLIP text-image score, and CLIP aesthetic scores for our method and other baselines. We also report the performance of Stable Diffusion without segmentation conditions. Methods marked with "*" are trained from scratch.

Condition reconstruction and FID score. We use the test set of ADE20K [96] to evaluate the conditioning fidelity. The state-of-the-art segmentation method OneFormer [35] achieves an Intersection-over-Union (IoU) with 0.58 on the ground-truth set. We use different methods to generate images with ADE20K segmentations and then apply One-

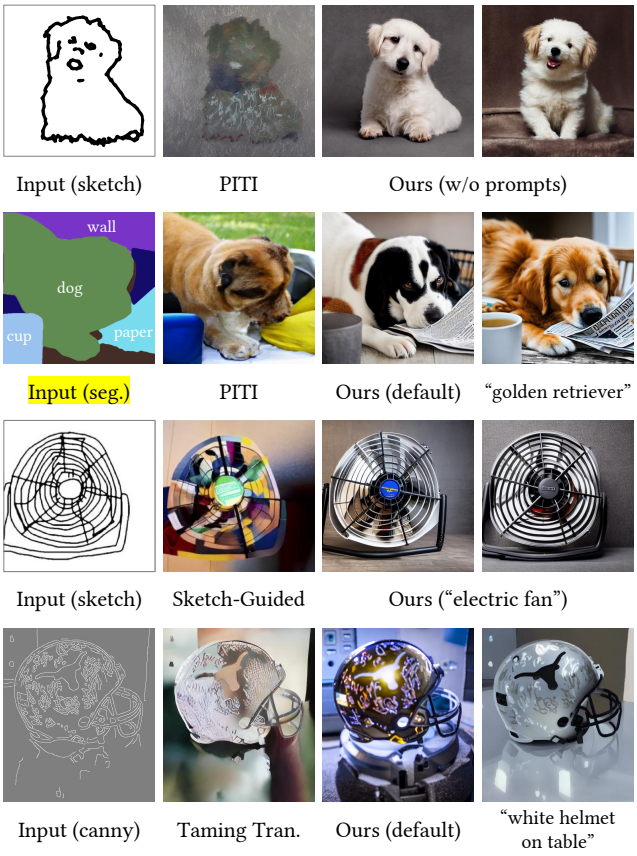

Figure 9: Comparison to previous methods.We present the qualitative comparisons to PITI [89], Sketch-Guided Diffusion [88], and Taming Transformers [19].

Former to detect the segmentations again to compute the reconstructed IoUs (Table 2). Besides, we use Frechet Inception Distance (FID) [28] to measure the distribution distance over randomly generated 512×512 image sets using different segmentation-conditioned methods, as well as text-image CLIP scores [66] and CLIP aesthetic score [79] in Table 3. See also the supplementary material for detailed settings.

4.4. Comparison to Previous Methods

Figure 9 presents a visual comparison of baselines and our method (Stable Diffusion + ControlNet). Specifically, we show the results of PITI [89], Sketch-Guided Diffusion [88], and Taming Transformers [19]. (Note that the backbone of PITI is OpenAI GLIDE [57] that have different visual quality and performance.) We observe that ControlNet can robustly handle diverse conditioning images and achieves sharp and clean results.

4.5. Discussion

Influence of training dataset sizes. We demonstrate the robustness of the ControlNet training in Figure 10. The training does not collapse with limited 1k images, and allows

Figure 10: The influence of different training dataset sizes. See also the supplementary material for extended examples.

Figure 11: Interpreting contents. If the input is ambiguous and the user does not mention object contents in prompts, the results look like the model tries to interpret input shapes.

Figure 12: Transfer pretrained ControlNets to community models [16, 61] without training the neural networks again.

the model to generate a recognizable lion. The learning is scalable when more data is provided.

Capability to interpret contents. We showcase ControlNet's capability to capture the semantics from input conditioning images in Figure 11.

Transferring to community models. Since ControlNets do not change the network topology of pretrained SD models, it can be directly applied to various models in the stable diffusion community, such as Comic Diffusion [61] and Protogen 3.4 [16], in Figure 12.

5. Conclusion

ControlNet is a neural network structure that learns conditional control for large pretrained text-to-image diffusion models. It reuses the large-scale pretrained layers of source models to build a deep and strong encoder to learn specific conditions. The original model and trainable copy are connected via “zero convolution” layers that eliminate harmful noise during training. Extensive experiments verify that ControlNet can effectively control Stable Diffusion with single or multiple conditions, with or without prompts. Results on diverse conditioning datasets show that the ControlNet struc-

ture is likely to be applicable to a wider range of conditions, and facilitate relevant applications.

Acknowledgment

This work was partially supported by the Stanford Institute for Human-Centered AI and the Brown Institute for Media Innovation.

References

[1] Sadia Afrin. Weight initialization in neural network, inspired by andrew ng, https://medium.com/@safrin1128/weight-initialization-in-neural-network-inspired-by-andrew-ng-e0066dc4a566, 2020. 3

[2] Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 7319–7328, Online, Aug. 2021. Association for Computational Linguistics. 3

[3] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Only a matter of style: Age transformation using a style-based regression model. *ACM Transactions on Graphics (TOG)*, 40(4), 2021. 3

[4] Yuval Alaluf, Omer Tov, Ron Mokady, Rinon Gal, and Amit Bermano. Hyperstyle: Stylegan inversion with hypernetworks for real image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18511–18521, 2022. 2

[5] Alembics. Disco diffusion, https://github.com/alembics/disco-diffusion, 2022. 3

[6] Omri Avrahami, Thomas Hayes, Oran Gafni, Sonal Gupta, Yaniv Taigman, Devi Parikh, Dani Lischinski, Ohad Fried, and Xi Yin. Spatext: Spatio-textual representation for controllable image generation. *arXiv preprint arXiv:2211.14305*, 2022. 2, 3

[7] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18208–18218, 2022. 3

[8] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. *arXiv preprint arXiv:2302.08113*, 2023. 3

[9] Dina Bashkirova, Jose Lezama, Kihyuk Sohn, Kate Saenko, and Irfan Essa. Masksketch: Unpaired structure-guided masked image generation. *arXiv preprint arXiv:2302.05496*, 2023. 3

[10] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022. 2, 3

[11] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6):679–698, 1986. 6

[12] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 6

[13] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12299–12310, 2021. 3

[14] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. *International Conference on Learning Representations*, 2023. 2

[15] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8789–8797, 2018. 3

[16] darkstorm2150. Protogen x3.4 (photorealism) official release, https://civitai.com/models/3666/protogen-x34-photorealism-official-release, 2022. 8

[17] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 3

[18] Tan M. Dinh, Anh Tuan Tran, Rang Nguyen, and Binh-Son Hua. Hyperinverter: Improving stylegan inversion via hypernetwork. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11389–11398, 2022. 2

[19] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12873–12883, 2021. 3, 5, 7, 8

[20] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. In *European Conference on Computer Vision (ECCV)*, pages 89–106. Springer, 2022. 2, 3

[21] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 2, 3

[22] Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022. 3

[23] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021. 2

[24] Geonmo Gu, Byungsoo Ko, SeoungHyun Go, Sung-Hyun Lee, Jingeun Lee, and Minchul Shin. Towards light-weight and real-time line segment detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022. 6

[25] David Ha, Andrew M. Dai, and Quoc V. Le. Hypernetworks. In *International Conference on Learning Representations*, 2017. 2

[26] Heathen. Hypernetwork style training, a tiny guide, stable-diffusion-webui, https://github.com/automatic1111/stable-diffusion-webui/discussions/2670, 2022. 2

[27] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 3

[28] 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. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 8

[29] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. 5

[30] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799, 2019. 2

[31] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 2

[32] Lianghua Huang, Di Chen, Yu Liu, Shen Yujun, Deli Zhao, and Zhou Jingren. Composer: Creative and controllable image synthesis with composable conditions. 2023. 3

[33] Nisha Huang, Fan Tang, Weiming Dong, Tong-Yee Lee, and Changsheng Xu. Region-aware diffusion for zero-shot text-driven image editing. *arXiv preprint arXiv:2302.11797*, 2023. 3

[34] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1125–1134, 2017. 1, 3

[35] Jitesh Jain, Jiachen Li, MangTik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. OneFormer: One Transformer to Rule Universal Image Segmentation. 2023. 7

[36] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *International Conference on Learning Representations*, 2018. 3

[37] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 3

[38] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *IEEE Transactions on Pattern Analysis*, 2021. 3

[39] Oren Katzir, Vicky Perepelook, Dani Lischinski, and Daniel Cohen-Or. Multi-level latent space structuring for generative control. *arXiv preprint arXiv:2202.05910*, 2022. 3

[40] 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. *arXiv preprint arXiv:2210.09276*, 2022. 3

[41] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2426–2435, 2022. 3

[42] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in Neural Information Processing Systems*, 34:21696–21707, 2021. 3

[43] Kurumuz. Novelai improvements on stable diffusion, https://blog.novelai.net/novelai-improvements-on-stable-diffusion-e10d38db82ac, 2022. 2

[44] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, May 2015. 3

[45] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 3

[46] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. *Proceedings of the 35th International Conference on Machine Learning*, 2018. 3

[47] Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. Measuring the intrinsic dimension of objective landscapes. *International Conference on Learning Representations*, 2018. 3

[48] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. 2023. 3

[49] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. *arXiv preprint arXiv:2203.16527*, 2022. 2

[50] Yanghao Li, Saining Xie, Xinlei Chen, Piotr Dollar, Kaiming He, and Ross Girshick. Benchmarking detection transfer learning with vision transformers. *arXiv preprint arXiv:2111.11429*, 2021. 2

[51] Arun Mallya, Dillon Davis, and Svetlana Lazebnik. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *European Conference on Computer Vision (ECCV)*, pages 67–82, 2018. 2

[52] Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7765–7773, 2018. 2

[53] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2021. 3

[54] Midjourney. https://www.midjourney.com/, 2023. 1, 3

[55] Ron Mokady, Omer Tov, Michal Yarom, Oran Lang, Inbar Mosseri, Tali Dekel, Daniel Cohen-Or, and Michal Irani. Self-distilled stylegan: Towards generation from internet photos. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–9, 2022. 3

[56] Chong Mou, Xintao Wang, Liangbin Xie, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning

adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453*, 2023. 2, 3

[57] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. GLIDE: towards photorealistic image generation and editing with text-guided diffusion models. *CoRR*, 2021. 8

[58] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. 2022. 3

[59] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 3

[60] Yotam Nitzan, Kfir Aberman, Qiurui He, Orly Liba, Michal Yarom, Yossi Gandelsman, Inbar Mosseri, Yael Pritch, and Daniel Cohen-Or. Mystyle: A personalized generative prior. *arXiv preprint arXiv:2203.17272*, 2022. 3

[61] ogkalu. Comic-diffusion v2, trained on 6 styles at once, https://huggingface.co/ogkalu/comic-diffusion, 2022. 8

[62] OpenAI. Dall-e-2, https://openai.com/product/dall-e-2, 2023. 1, 3

[63] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2337–2346, 2019. 3

[64] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. *arXiv preprint arXiv:2302.03027*, 2023. 3

[65] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2085–2094, October 2021. 3

[66] 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. 2, 3, 4, 8

[67] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 3

[68] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 3

[69] 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. 6

[70] Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. Efficient parametrization of multi-domain deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8119–8127, 2018. 2

[71] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 3

[72] 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, 2, 3, 4, 5, 7

[73] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention MICCAI International Conference*, pages 234–241, 2015. 4

[74] Amir Rosenfeld and John K Tsotsos. Incremental learning through deep adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(3):651–663, 2018. 2

[75] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022. 2, 3

[76] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, Oct. 1986. 3

[77] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, SIGGRAPH '22, New York, NY, USA, 2022. Association for Computing Machinery. 3

[78] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 3

[79] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 2, 8

[80] Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *International Conference on Machine Learning*, pages 4548–4557. PMLR, 2018. 2

[81] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using

nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 3

[82] Stability. Stable diffusion v1.5 model card, https://huggingface.co/runwayml/stable-diffusion-v1-5, 2022. 2, 3

[83] Stability. Stable diffusion v2 model card, stable-diffusion-2-depth, https://huggingface.co/stabilityai/stable-diffusion-2-depth, 2022. 3, 7

[84] Asa Cooper Stickland and Iain Murray. Bert and pals: Projected attention layers for efficient adaptation in multi-task learning. In *International Conference on Machine Learning*, pages 5986–5995, 2019. 2

[85] Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. Vl-adapter: Parameter-efficient transfer learning for vision-and-language tasks. *arXiv preprint arXiv:2112.06825*, 2021. 2

[86] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. *arXiv preprint arXiv:2211.12572*, 2022. 3

[87] 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. 6

[88] Andrey Voynov, Kfir Abernan, and Daniel Cohen-Or. Sketch-guided text-to-image diffusion models. 2022. 3, 6, 7, 8

[89] Tengfei Wang, Ting Zhang, Bo Zhang, Hao Ouyang, Dong Chen, Qifeng Chen, and Fang Wen. Pretraining is all you need for image-to-image translation. 2022. 3, 6, 7, 8

[90] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8798–8807, 2018. 3

[91] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1395–1403, 2015. 6

[92] Jeffrey O. Zhang, Alexander Sax, Amir Zamir, Leonidas J. Guibas, and Jitendra Malik. Side-tuning: Network adaptation via additive side networks. In *European Conference on Computer Vision (ECCV)*, pages 698–714. Springer, 2020. 2

[93] Pan Zhang, Bo Zhang, Dong Chen, Lu Yuan, and Fang Wen. Cross-domain correspondence learning for exemplar-based image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5143–5153, 2020. 3

[94] Renrui Zhang, Rongyao Fang, Peng Gao, Wei Zhang, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*, 2021. 2

[95] Jiawei Zhao, Florian Schäfer, and Anima Anandkumar. Zero initialization: Initializing residual networks with only zeros and ones. *arXiv*, 2021. 3

[96] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 633–641, 2017. 6, 7

[97] Xingran Zhou, Bo Zhang, Ting Zhang, Pan Zhang, Jianmin Bao, Dong Chen, Zhongfei Zhang, and Fang Wen. Cocosnet v2: Full-resolution correspondence learning for image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11465–11475, 2021. 3

[98] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017. 1, 3

[99] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. *Advances in Neural Information Processing Systems*, 30, 2017. 3