Adapting Diffusion Models for Improved Prompt Compliance and Controllable Image Synthesis

Deepak Sridhar   **Abhishek Peri**   **Rohith Rachala**   **Nuno Vasconcelos**

Department of Electrical and Computer Engineering
University of California, San Diego
{desridha, aperi, rrachala, nvasconcelos}@ucsd.edu

Abstract

Recent advances in generative modeling with diffusion processes (DPs) enabled breakthroughs in image synthesis. Despite impressive image quality, these models have various prompt compliance problems, including low recall in generating multiple objects, difficulty in generating text in images, and meeting constraints like object locations and pose. For fine-grained editing and manipulation, they also require fine-grained semantic or instance maps that are tedious to produce manually. While prompt compliance can be enhanced by addition of loss functions at inference, this is time consuming and does not scale to complex scenes. To overcome these limitations, this work introduces a new family of *Factor Graph Diffusion Models* (FG-DMs) that models the joint distribution of images and conditioning variables, such as semantic, sketch, depth or normal maps via a factor graph decomposition. This joint structure has several advantages, including support for efficient sampling based prompt compliance schemes, which produce images of high object recall, semi-automated fine-grained editing, text-based editing of conditions with noise inversion, explainability at intermediate levels, ability to produce labeled datasets for the training of downstream models such as segmentation or depth, training with missing data, and continual learning where new conditioning variables can be added with minimal or no modifications to the existing structure. We propose an implementation of FG-DMs by adapting a pre-trained Stable Diffusion (SD) model to implement all FG-DM factors, using only COCO dataset, and show that it is effective in generating images with 15% higher recall than SD while retaining its generalization ability. We introduce an attention distillation loss that encourages consistency among the attention maps of all factors, improving the fidelity of the generated conditions and image. We also show that training FG-DMs from scratch on MM-CelebA-HQ, Cityscapes, ADE20K, and COCO produce images of high quality (FID) and diversity (LPIPS). **Project Page:** FG-DM

1 Introduction

Diffusion models (DMs) (44; 15; 10; 41) have recently shown great promise for image synthesis and popularized text-to-image (T2I) synthesis, where an image is generated in response to a text prompt. However, T2I synthesis offers limited control over image details. Even models trained at scale, such as Stable Diffusion (SD) (41) or DALL·E 2 (37), have significant prompt compliance problems, such as difficulty in generating multiple objects (7; 34), difficulty in generating text in images (26; 46), or to consistently produce images under certain spatial constraints, like object locations and poses (56; 6; 1). These limitations have been addressed through two main lines of research. One possibility is to use *inference-based prompt-compliance* (IBPC) methods (7; 34; 40), which use loss functions that operate on the cross-attention maps between image and prompt tokens to improve prompt compliance

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

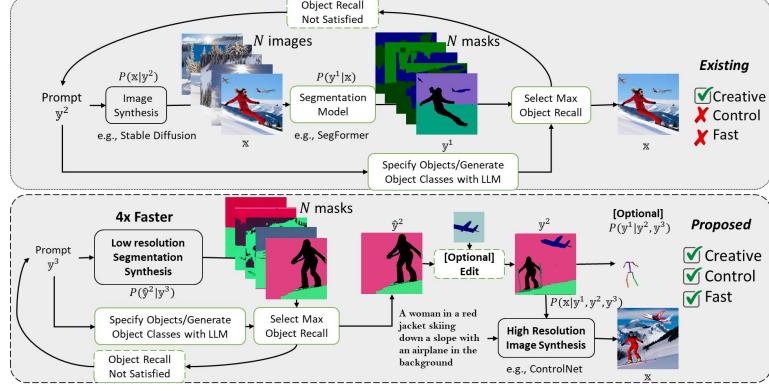



Figure 1: Comparison of FG-DM (bottom) against Stable Diffusion (top) for sampling images with high object recall by modeling the joint distribution of images and conditioning variables. FG-DM supports creative, controllable, interpretable and faster (4x) image synthesis than Stable Diffusion to achieve the desired object recall. Note that the conditions $\mathbf{y^1}$ or $\mathbf{y^2}$ can be null due to classifier-free guidance training.

at inference. While these methods are effective for prompts involving a small number of objects, they tend to underperform for prompts involving complex scenes. Furthermore, because their complexity grows linearly with the number of scene objects, they tend to be prohibitively time-consuming for such scenes. A second possibility is to rely on DMs that support *visual conditioning,* in the form of sketches (33), bounding boxes (9), scene graphs (53), reference images (52), etc. *Visually conditioned DMs* (VC-DMs) are usually extensions of T2I-DMs trained at scale. For example, ControlNet (58), T2I-Adapter (29) and Uni-ControlNet (60) use a learnable branch to modulate the features of a pre-trained SD model according to a visual condition. Despite their success, VC-DMs have important limitations, inherent to *models of the conditional distribution* $P(\mathbf{x}|\{\mathbf{y}^i\})$ of image $\mathbf{x}$ given conditions $\mathbf{y}^i$: the need for user supplied conditions. The manual specification of visual conditions, like segmentation masks or normal maps, requires users with considerable time and skill. While, as illustrated in Figure 1 (top), conditions $\mathbf{y}^i$ can be extracted from existing images, this requires additional vision models (e.g. segmentation or edge detection), which is time-consuming.

In this work, we consider an alternative framework that attempts to mitigate all these problems using a simple but unexplored prompt compliance scheme that we denote as *sampling-based prompt compliance* (SBPC). The idea, illustrated in the top of Figure 1 is to sample a batch of N images using different DM seeds, relying on an external model (e.g. segmentation) to measure prompt compliance (e.g. by measuring object recall) and choosing the image that best complies with the prompt. While this strategy is frequently successful, generating multiple high resolution images significantly increases the inference time, rendering the approach impractical even for small values of N as we will show in section 4. Furthermore, it does not address the need for specification of the conditions $\mathbf{y}^i$ required by the VC-DM. We address these problems by introducing a new family of *Factor Graph-DMs* (FG-DMs). As illustrated in the bottom of Figure 1, a FG-DM is a modular implementation of the joint distribution $P(\mathbf{x}, \{\mathbf{y}^i\})$ by decomposing the image synthesis into two or more factors, that are implemented by jointly trained VC-DMs. The figure shows an example decomposition of the distribution $P(\mathbf{x}, \{\mathbf{y}^i\}_{i=1}^2|\mathbf{y}^3)$ of image $\mathbf{x}$, pose $\mathbf{y}^1$, and segmentation $\mathbf{y}^2$, given prompt $\mathbf{y}^3$, into three factors: $P(\mathbf{y}^2|\mathbf{y}^3)$ for the synthesis of segmentation given prompt, $P(\mathbf{y}^1|\{\mathbf{y}^i\}_{i=2}^3)$ for pose $\mathbf{y}^1$ synthesis conditioned on both, and $P(\mathbf{x}|\{\mathbf{y}^i\}_{i=1}^3)$ for image synthesis given all conditions.

The FG-DM framework has several advantages. First, prompt compliance can usually be measured (e.g. by computing object recall) by inspecting the conditions $\mathbf{y}_i$ (e.g. segmentation map). The gain is that these can be generated with less diffusion steps and resolution than the final image $\mathbf{x}$. For example, we have observed no loss of image quality by sampling segmentation maps of quarter resolution. This increases the speed of SBPC **by 4x**, making it a practical prompt compliance scheme. We show that sampling with $N = 10$ different seeds and choosing the image of maximum recall increases prompt compliance (object recall) by 15% as compared to sampling with one seed. For complex scenes, it is also much faster and more effective than using IBPC methods (see Table 4). Second, as illustrated on the bottom of Figure 1, the modular nature of the FG-DM offers image editing capabilities. New objects can be added by *synthesizing* them separately while existing objects can be resized and/or moved to the desired spatial location. In Figure 1 (also in Figure 2 with more

2

Figure 2: FG-DM-based controllable image generation via editing segmentation, depth and sketch maps. Top: generated conditions and images. Bottom: edited ones. Note that only the segmentation map is edited, pose and images are conditionally generated given edited map.

Figure 3: Synthesized segmentation/depth/sketch/normal maps and corresponding images by an FG-DM adapted from SD using COCO. The FG-DM generalizes to prompts beyond this dataset such as porcupine, chimp and other creative prompts shown.

detail), an airplane is added to the background while the person is resized, moved to the left of the image, and pose flipped. We introduce a simple image editing tool for performing these edits. Figure 2 shows other examples of fine-grained image editing with FG-DM for semantic, depth, and sketch maps factors. In the center, the dog is placed behind the sandcastle (and some objects are added to the foreground) by manipulation of a depth map, and in the right the desired text "Hello FG-DM" is scribbled on the sketch map produced by the model. Third, the FG-DM can reuse factors in the literature. For example, the ControlNet is used to implement the image synthesis factor $P(\mathbf{x}|\{\mathbf{y}^i\}_{i=1}^3)$ in Figures 1 and 2. Fourth, the FG-DM can produce labeled datasets for the training of downstream systems (e.g. image segmentation), and naturally supports continual learning schemes, where image synthesis and manipulation are gradually enhanced by the progressive addition of new VC-DMs to the factor graph, with limited or no retraining of existing ones.

Since FG-DM models the joint distribution $P(\mathbf{x}, \{\mathbf{y}^i\})$, it is a variant of *joint DMs* (JDMs) and training a FG-DM from scratch requires large scale datasets of (condition, image) pairs, which are expensive to obtain. However, we show that this difficulty can be overcome by adapting existing foundation VC-DMs, such as SD, to implement each factor of the FG-DM. We propose a joint prompting scheme to implement this adaptation and introduce an attention distillation loss that distills the attention maps from a pre-trained SD model to implement the condition synthesis factors $P(\mathbf{y}^k|\{\mathbf{y}^i\}_{i=1}^{k-1})$, by minimizing the KL divergence between the two and show that this improves the fidelity of the generated conditions. This greatly reduces the training costs and enables much greater generalization ability than would be possible by training on existing (condition,image) datasets alone. Figure 3 shows that FG-DM exhibits robust generalization by synthesizing depth, normals and their corresponding images for novel objects not present in the training data. This approach also facilitates cross-model information transfer, enhancing explainability and showcasing the FG-DM's versatility in complex synthesis tasks. In summary, this paper makes the following contributions

- We propose a new framework for T2I, the FG-DM, which supports the modeling of the joint distribution of images and conditioning variables, while maintaining access to all their conditional relationships
- We show that FG-DMs enable practical SBPC, by leveraging fast condition synthesis and filtering by object recall, and allow both fine-grained image editing with minimal effort and data augmentation.

3

- We show that FG-DMs can be implemented by adapting pre-trained T2I models (e.g., SD) using efficient prompt-based adaptation on relatively small datasets (e.g. COCO) while exhibiting interesting generalization, e.g. generalizing to prompts involving concepts not covered by these datasets.
- We introduce an attention distillation loss that improves the fidelity of the synthesized conditions and enables transfer of information from SD to all factors of the FG-DM.
- We show that FG-DMs trained from scratch on domain-specific datasets such as MM-CelebA-HQ, ADE20K, Cityscapes and COCO consistently obtains high quality images (lower FID scores) with higher image diversity, i.e. higher LPIPS scores, than standard JDMs.

2 Related Work

Text-to-Image (T2I) Diffusion models(16; 30; 37; 38) learn to synthesize images from noise conditioned by a text encoder, usually CLIP (36). Latent DMs (LDMs) (41) implement DMs in the latent space learned by an autoencoder trained on a very large image dataset, to reduce inference and training costs. T2I models typically employ classifier-free guidance (17) to balance prompt compliance with sample diversity. However, they often struggle with complex prompts, which led to the development of conditional DMs. They model the distribution $P(\mathbf{x}|\{\mathbf{y}^k\})$ of image $\mathbf{x}$ given a set of K conditions $\mathbf{y}^k$. VC-DMs, such as ControlNet (58), T2I-Adapter (29), and HumanSD (19), use adapters to condition image generation on visual variables $\mathbf{y}^i$, such as depth, semantic, pose, or normal maps. However, these methods are limited to conditions $\mathbf{y}^k$ generated from existing images or by manual sketching, which can be hard to obtain, especially for $K > 1$ (e.g. simultaneous segmentation and surface normals), and may be inconsistent. Instead, the proposed FG-DM enables the automated joint generation of all conditions while still allowing users the ability to edit them.

Joint Models model the distribution $P(\mathbf{x}, \{\mathbf{y}^k\})$, frequently by concatenating all variables during image generation. For example, SemanticGAN (2) and GCDP (32) use a single model to generate pairs of images $\mathbf{x}$ and semantic maps $\mathbf{y}$, while Hyper Human (24) uses a single model to generate depth maps, normal maps, and images from human pose skeletons. These models lack access to the conditional distribution $P(\mathbf{x}|\mathbf{y})$, which is critical for fine-grained image editing. 4M (27) trains an autoregressive generalist model using all conditioning variables jointly. However, it is not modular, requires large paired datasets to train and does not support continual learning of new classes. FG-DMs are more closely related to methods like Semantic Bottleneck GAN (2), Make-a-Scene (12) and Semantic Palette (21), which model the joint distribution as the composition of a semantic map distribution $P(\mathbf{y})$ and a conditional model $P(\mathbf{x}|\mathbf{y})$ for generating image $\mathbf{x}$ given synthesized layout $\mathbf{y}$. "Make a Scene" (12) learns a VAE from segmentation maps and samples segmentation tokens from its latent space. A transformer then combines those with tokens derived from text, to synthesize an image. All joint models above have important limitations: they are trained from scratch, only consider semantic (discrete) conditioning variables and do not scale to the generation of high-fidelity images of complex natural scenes, involving a large number of semantic classes (32), such as those in COCO (23) without access to large-scale datasets.

Inference-Based Prompt Compliance (IBPC) methods(7; 34; 40) attempt to improve prompt compliance by optimizing the noise latent at each diffusion iteration with a loss that maximizes attention to each noun or the binding between prompt attributes and nouns. However, these methods are time-consuming, require careful hyperparameter fine-tuning and do not work well for multiple object scenes. The proposed FG-DMs build on the power of VC-DMs and rely on SBPC, which samples various images and selects the one most compliant with the prompt. Since this only requires the synthesis of conditions, it can be done efficiently even for complex scenes.

3 The Factor-Graph Diffusion Model

3.1 Diffusion Models

DMs: DMs (44; 15) are probabilistic models based on two Markov chains. In the forward direction, white Gaussian noise is recursively added to image $\mathbf{x}$, according to

$$\mathbf{z}_t = \sqrt{\alpha_t}\mathbf{z}_0 + \sqrt{1 - \alpha_t}\epsilon_t, \quad \epsilon_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{1}$$

4

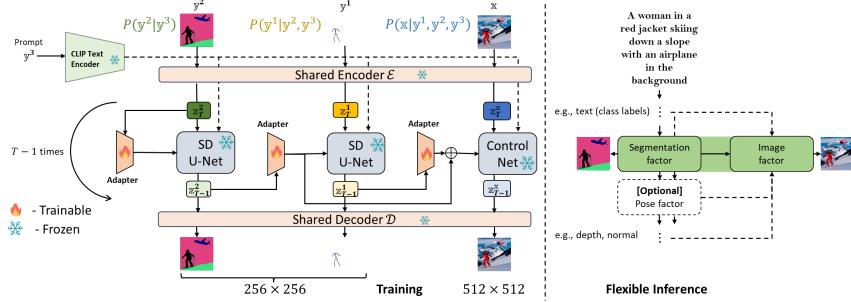



Figure 4: **Left:** Training of FG-DM for distribution $P(\mathbf{x}, \mathbf{y}^1, \mathbf{y}^2|\mathbf{y}^3)$ of image $\mathbf{x}$, segmentation mask $\mathbf{y}^2$, and pose map $\mathbf{y}^1$, given text prompt $\mathbf{y}^3$. Each factor (conditional probability written at top of each figure) is implemented by adapting a pretrained SD model to generate a visual condition. The SD model is frozen and only a small adapter is learned per factor. The final (image generation) factor uses ControlNet without adaptation. The encoder-decoder pair and SD backbone are shared among all factors, reducing the total number of parameters. Conditional generation chains are trained at lower resolution for better inference throughput. **Right:** The FG-DM offers a flexible inference framework due to classifier-free guidance training, where only a desired subset of the factors are run, as shown in the highlighted green area.

where $\mathbf{z}_0 = \mathbf{x}$, $\mathcal{N}$ is the normal distribution, $\mathbf{I}$ the identity matrix, $\alpha_t = \prod_{k=1}^{t}(1 - \beta_k)$, and β_t a pre-specified variance. In the reverse process, a neural network $\epsilon_\theta(\mathbf{z}_t, t)$ recurrently denoises $\mathbf{z}_t$ to recover $\mathbf{x}$. This network is trained to predict noise ϵ_t, by minimizing the risk defined by the loss $\mathcal{L} = ||\epsilon_t - \epsilon_\theta(\mathbf{z}_t, t)||^2$. Samples are generated with $\mathbf{z}_{t-1} = f(\mathbf{z}_t, \epsilon_\theta(\mathbf{z}_t, t))$ where

$$f(\mathbf{z}_t, \epsilon_\theta) = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{z}_t - \frac{\beta_t}{\sqrt{1 - \alpha_t}} \epsilon_\theta \right) + \sigma\xi, \tag{2}$$

with $\xi \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \mathbf{z}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The network $\epsilon_\theta(\mathbf{z}_t, t)$ is usually a U-Net (42) with attention (47).

3.2 The FG-DM model

The FG-DM is a conceptually simple generalization of the DM to support K conditioning variables $\mathbf{y}^i$. Rather than the conditional distribution $P(\mathbf{x}|\{\mathbf{y}^i\})$ it models the *joint* $P(\mathbf{x}, \{\mathbf{y}^i\})$. Prior joint models (32; 24) use a single joint denoising U-Net in the pixel or latent space to jointly synthesize $\mathbf{x}$ and $\{\mathbf{y}^i\}$. This limits scalability to multiple conditions, increases the difficulty of editing the synthesized $\mathbf{x}$, and requires retraining the entire model to add new conditions $\mathbf{y}$. The FG-DM instead leverages the decomposition of the joint into a factor graph (11) composed by a sequence of conditional distributions, or factors, according to

$$P(\mathbf{x}, \{\mathbf{y}^i\}) = P(\mathbf{x}|\{\mathbf{y}^i\}_{i=1}^{K})P(\mathbf{y}^1|\{\mathbf{y}^i\}_{i=2}^{K}) \cdots P(\mathbf{y}^K), \tag{3}$$

where there are usually conditional independence relations that simplify the terms on the right hand side. In any case, (3) enables the implementation of the joint DM as a modular composition of conditional DMs. This is illustrated in Figure 4, which shows the FG-DM discussed in Figure 1.

Synthesis of conditioning variables. We convert all conditioning variables to 3-channel inputs and use the pre-trained $\mathcal{E}$-$\mathcal{D}$ pair from the SD model to map them into the latent codes for efficient training. See appendix section A.1.1 for more details on this process.

Sampling: The FG-DM samples $\mathbf{y}^i$ and $\mathbf{x}$ as follows. Let ϵ_{θ^x} be a DM for $P(\mathbf{x}|\{\mathbf{y}^i\})$, and ϵ_{θ^i} a DM for $P(\mathbf{y}^i|\mathbf{y}^{i+1}, \cdots, \mathbf{y}^K)$. In the forward direction, $\mathbf{z}_t^x$ and $\mathbf{z}_t^i$, the noisy versions of $\mathbf{x}$ and $\mathbf{y}^i$, respectively, are generated by using (1) with $\mathbf{z}_0^x = \mathbf{x}$ and $\mathbf{z}_0^i = \mathbf{y}^i$. In the reverse process, each denoising step is implemented with

$$\mathbf{z}_{t-1}^K = f(\mathbf{z}_t^K, \epsilon_{\theta^K}(\mathbf{z}_t^K, t)), \tag{4}$$

$$\mathbf{z}_{t-1}^i = f(\mathbf{z}_t^i, \epsilon_{\theta^i}(\mathbf{z}_t^i, \ldots, \mathbf{z}_{t-1}^K, t)), \forall i < K \tag{5}$$

$$\mathbf{z}_{t-1}^x = f(\mathbf{z}_t^x, \epsilon_{\theta^x}(\mathbf{z}_t^x, \mathbf{z}_{t-1}^1, \ldots, \mathbf{z}_{t-1}^K, t)) \tag{6}$$

where $f(.)$ is the recursion of (2). All conditions are sampled at each denoising step.

5

3.3 Adaptation of pretrained DM

Architecture: To adapt a pretrained SD model into a FG-DM factor, we modify the T2I-Adapter (29) to be conditioned on the current timestep t and use encoded latent features of the condition(s) from previous factors as input to the adapter of the current factor. Figure 4 shows how the conditioning of (4)-(6) is implemented: noisy latent feature $\mathbf{z}_t^K$ is fed to the first adapter, and the denoised latents $\mathbf{z}_{t-1}^i$ of each VC-DM are fed to the adapters of the subsequent VC-DMs (denoising of $\mathbf{z}_t^k, k < i$). The adapter consists of four feature extraction blocks with one convolution layer and two timestep-residual blocks for each scale. The encoder features $F_{i,t}^{\text{enc}}$ at the output of U-Net block i are modulated as $\hat{F}_{i,t}^{\text{enc}} = F_{i,t}^{\text{enc}} + F_{i,t}^c, \quad i \in \{1, 2, 3, 4\}$. where $F_{i,t}^c$ are the features produced by an adapter branch associated with $F_{i,t}^{\text{enc}}$ at timestep t. The SD model is kept frozen, only the adapter branches are learned per factor model.

Training with Classifier-Free Guidance (CFG): Given a training example with all the conditioning variables $(\mathbf{x}_j, \mathbf{y}_j^1, \ldots, \mathbf{y}_j^K)$, we randomly select $1, \cdots, K-1$ conditioning variables as null condition for CFG, 20% of the training time. This facilitates unconditional training of each factor model which supports flexible inference. As a result, only a desired subset of the conditions of the FG-DM are run, as illustrated in the right of Figure 4 (highlighted in green).

Attention Distillation. The synthesis of conditions like semantic or normal maps requires learning to precisely associate spatial regions with object identities or geometry, which is a difficult task. To encourage the binding of these properties across conditions and image, we ground them on the SD attention maps, which are known to encode prompt semantics (13). The intuition is that the attention dot-products between word tokens and image regions should remain approximately constant for all factor models. Ideally, a given text prompt word should elicit a similar region of attention in the VC-DM that synthesizes each condition. To encourage this, we introduce an attention distillation loss, based on the KL divergence between the self and cross attention maps of the pretrained SD model and those of the adapted models. This can be viewed as using the pretrained DM as a teacher that distills the knowledge contained in the attention maps to the adapted student. This distillation also helps the adapted model retain the generalization to unseen text prompts, such as the examples of Fig. 1. Formally, the attention distillation loss is defined as

$$\mathcal{L}_{KL}(f^t, f^s) = \sum_{j \in \{0,1\}} \text{KL}\left(\sum_{i=1}^{L_j} g_{ij}(F_{ij}^t(Q_{ij}^t, K)) \,\Big\|\, \sum_{i=1}^{L_j} g_{ij}(F_{ij}^s(Q_{ij}^s, K))\right) = \sum_{j \in \{0,1\}} \sum_{i=1}^{L_j} g_{ij}(F_{ij}^t(Q_{ij}^t, K)) \log\left(\frac{\sum_{i=1}^{L_j} g_{ij}(F_{ij}^t(Q_{ij}^t, K))}{\sum_{i=1}^{L_j} g_{ij}(F_{ij}^s(Q_{ij}^s, K))}\right) \quad (7)$$

where superscript t (s) denotes SD-teacher (SD-student), g implements a bilinear interpolation needed to upscale all maps to a common size, F is the softmax of the products between query noise feature matrix (Q) and key CLIP text feature (K), L_j number of attention layers, and index $j = 0, 1$ denotes self and cross attention layers respectively. The overall loss is the sum of distillation losses between teacher t (pre-trained SD model) and all students $\mathcal{L}_{KL} = \sum_{i=1}^K \mathcal{L}_{KL}(f^t, f^{s^i})$, where s^i is the student model adapted to condition $\mathbf{y}^i$. For multi-condition FG-DMs, the distillation loss is only required for the synthesis of the first condition, which is conditioned by text alone. For the subsequent factors, which are already conditioned by a visual condition (e.g. pose conditioned by segmentation in Figure 4), the attention distillation loss is not needed.

Loss: Since the noise introduced in the different factors is independent, the networks are optimized to minimize the risk defined by the loss

$$\mathcal{L}_{FG} = ||\epsilon_t^x - \epsilon_{\theta^x}(\mathbf{z}_t^x, \mathbf{z}_{t-1}^1, \ldots, \mathbf{z}_{t-1}^K, t)||^2 + \sum_{i=1}^K ||\epsilon_t^i - \epsilon_\theta^i(\mathbf{z}_t^i, \ldots, \mathbf{z}_{t-1}^K, t)||^2 + \lambda_{KL}\mathcal{L}_{KL} \quad (8)$$

Training from scratch: The FG-DM can also be trained from scratch by simply concatenating the latent representation of the previous condition(s) and noise to generate the next condition instead of using adapters. Please refer to the appendix section A.1.2 for a detailed discussion.

4 Experimental Results

Datasets and models: We consider four conditioning variables in this paper: segmentation, depth, normal and sketch maps. The pretrained SD v1.4 model is adapted using the COCO-WholeBody dataset(23; 18), with 256 input resolution, to train all condition factors. Groundtruth (GT) is as

6

| "An image of a small rhino" | "An image of a giant squirrel" | "A man eating a hotdog" | "A picture of a volcanic eruption" | "A picture of a sungod" | "A penguin riding a bicycle" | "A man with a hat and glasses" | "A bird with headphones singing in front of a microphone" | "An older woman sitting near a window of a train." | "A room with a fireplace and a television" |

Figure 5: **More qualitative results of FG-DM** to synthesize segmentation, depth, normal and sketch maps and their corresponding images. See appendix Figure. 14 for the higher resolution version.

follows: COCO GT segmentations, HED soft edge (51) for sketch maps, and off-the-shelf MIDAS (39; 3) detector for depth and normal maps. We also present results for an FG-DM trained from scratch on MM-CelebAMaskHQ (22), and for other datasets in appendix, where implementation details are also given.

Evaluation: Visual quality is evaluated with Frechet Inception Distance (FID) (14), image diversity with LPIPS (59). We also report the Precision/Recall Values of (20) and evaluate prompt alignment with CLIP score (CLIP-s) (36). All speeds are reported using a NVIDIA-A10 GPU.

Qualitative and Quantitative results: Figure 5 shows additional qualitative results of synthesized segmentation (columns 1-3), depth (columns 4-6), normal (columns 7-8), sketch (columns 9-10) maps and their corresponding images for the prompts shown below each image. The FG-DM leverages the generalization of the pre-trained SD model to synthesize segmentation maps for object classes such as rhino and squirrel, beyond the training set (COCO). The semantic maps are colored with unique colors, allowing the easy extraction of both object masks and class labels. This shows the potential of the FG-DM for open-set segmentation, e.g the synthesis of training data for segmentation models. Note that these results demonstrate a "double-generalization" ability. While the FG-DM was never trained on squirrels or rhinos, SD was never trained to produce segmentation masks. However, FG-DM adapted from SD produces segmentations for squirrels and rhinos. The fourth column shows the depth map and image synthesized for the prompt *"A picture of a volcanic eruption"*. In the fifth column the same caption is used to create the depth map, while the image is created with the prompt *"A picture of a sungod"*. This shows that even when there is a mismatch between the prompts used for different factors, the FG-DM is able to produce meaningful images. This is a benefit of the FG-DM *modularity*. Table 5 compares the text-to-image synthesis quality of SD, 4M-XL (28) and four FG-DM models. Despite using a bigger model and training from scratch, 4M-XL model generates lower quality images than SD and FG-DM as seen from the higher FID score. The FG-DM has higher image quality than SD for the segmentation and normal map conditions and higher clip score for sketch condition showing the effectiveness of adaptation.

User Study: We conducted a human evaluation to compare the qualitative performance of the FG-DM (adapted from SD) with $N = 1$ to the conventional combination of SD+CEM, where CEM is an external condition extraction model (CEM), for both segmentation and depth conditions. We collected 51 unique prompts, composed by a random subset of COCO validation prompts and a subset of creative examples. We sampled 51 (image,condition) pairs - 35 pairs of (image,depth map), 16 pairs of (image,segmentation map) - using the FG-DM. For SD+CEM, images were sampled with SD for the same prompts, and fed to a CEM implemented with MIDAS (3) for depth and OpenSeed (57) for segmentation. The study was performed on Amazon Mechanical Turk, using 10 unique expert human evaluators per image. These were asked to compare the quality of the pairs produced by the two approaches and vote for the best result in terms of prompt alignment and visual appeal. Table 1 shows that evaluators found FG-DM generated images (masks) to have higher quality 61.37% (63.13%) and better prompt alignment 57.68% (60.98%) of the time. These results show that the FG-DM produces images and masks that have higher quality and better prompt alignment.

Qualitative Image Editing Results: Figure 6 shows additional examples of synthetic image editing using the FG-DM. Diffusion models have well known difficulties to perform operations such as switching people's locations (5) or synthesizing images with text (26; 46). The first four columns show that the FG-DM is a viable solution to these problems. The first column shows the image

7

Table 1: User study on the qualitative preference of images/condition pairs generated by the FG-DM and SD+CEM, using 10 unique human evaluators. A. denotes (prompt) Adherence and Q. denotes Quality.

Model	Img A.↑	Cond A.↑	Img Q.↑	Cond Q.↑
No clear winner	4.11	4.70	2.90	3.33
SD+CEM	37.84	34.31	35.49	33.52
FG-DM	**57.84**	**60.98**	**61.37**	**63.13**

Table 2: Ablation of attention distillation loss for T2I synthesis on COCO validation for FG-DM. FID reported for Images/Conditions.

Model	Distill	FID ↓	P↑	R↑	CLIP-s ↑
Seg		20.9/164.8	0.54	0.49	28.5
Seg	✓	**19.6/159.2**	**0.56**	**0.52**	**28.5**
Normal		20.60/126.7	0.56	0.49	28.6
Normal	✓	**19.30/123.9**	**0.59**	**0.55**	**28.7**

Table 3: Object recall statistics for sampling FG-DM with different seeds and timesteps on the ADE20K validation set prompts.

Samples/Batch	Avg. Min. Recall (%)	Avg. Max.↑ Recall (%)	Avg. Med.↑ Recall (%)	Avg. # Imgs 0.5 0.75 0.9	Time↓ (s)
t=10,N=1	69	69	69	0.9 0.4 0.1	**0.45**
t=20,N=1	68	68	68	0.9 0.4 0.1	0.81
t=10,N=5	60	74	**70**	4.5 1.9 0.6	1.10
t=10,N=10	55	**75**	**70**	9.0 3.6 1.2	1.75
t=20,N=10	55	**75**	**70**	8.9 3.6 1.1	3.3

Table 4: Quantitative comparison of Object Recall for different models and configurations on the ADE20K validation set prompts.

Model	Clip-S↑	Avg. Recall↑	Time (s) ↓
SD (N=1,t=20)	**0.301**	59.8	**3.25**
A-E (N=1,t=20)	0.295	63.6	36.0
FG-DM (N=1,t=20)	0.295	67.8	4.81
SD (N=10,t=20)	**0.301**	75.1	23.0
FG-DM (N=10,t=10)	0.294	**75.1**	5.75 (4x↓)
FG-DM (N=10,t=20)	0.296	75.0	7.35 (3x↓)

Generated	Edited	Generated	Edited	Edited	Edited	Generated	Generated

"Two men are shaking hands" | "Two men facing away from one other" | "a dog wearing a hoodie saying DIFF'" | "An image of a giant squirrel beside a small rhino" | Stable Diffusion v1.4 | Stable Diffusion v1.5

Figure 6: **Examples of images generated by FG-DM after editing and comparison with popular text-to-image models.** Editing is shown for flipping persons (columns 1-2), writing the desired text (columns 3-4) or realizing a difficult prompt (columns 5-6). Images generated by Stable Diffusion v1.4 and v1.5 for the same prompt are shown in the last two columns.

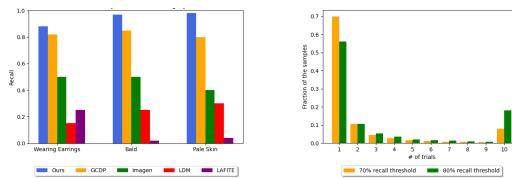



Figure 7: Attribute recall verification with FG-DM on MM-CelebA-HQ. **Left:** Semantic Attribute Recall. **Right:** Histogram of the number of trials needed to reach the specified recall.

Table 5: Quantitative results of Text-to-Image synthesis on COCO for FG-DM with segmentation, depth, sketch and normal conditions.

Model	FID ↓	P ↑	R ↑	CLIP-s ↑
SD	20.5	0.50	**0.69**	28.5
4M-XL Seg	39.1	0.49	0.30	28.8
FG-DM Seg	**19.6**	**0.56**	0.52	**28.5**
FG-DM Depth	27.0	0.45	0.52	27.6
FG-DM Sketch	25.3	0.39	0.61	**30.3**
FG-DM Normal	**19.3**	**0.59**	0.55	**28.7**

generated by FG-DM for two men shaking hands while the second shows the edited version where the two men are flipped, so as to face away from each other, and combined with a different background. The 3rd and 4th columns show an example where the user scribbles "DIFF" in the synthesized sketch map, which is then propagated to the image. The last four columns show examples of a difficult prompt, *"An image of a giant squirrel beside a small rhino"*, unlikely to be found on the web, for which existing T2I Models (SD v1.4/1.5) fail (columns 7-8). The FG-DM generates meaningful images (5th and 6th column) by simply editing the masks sampled by the segmentation factor, as discussed in Figure 1. In this example, the animal regions shown in Figure 5 (columns 1-2) were resized, flipped and pasted onto a common segmentation. Note how FG-DM allows precise control over object positions and orientations, which differ from those of Figure 5.

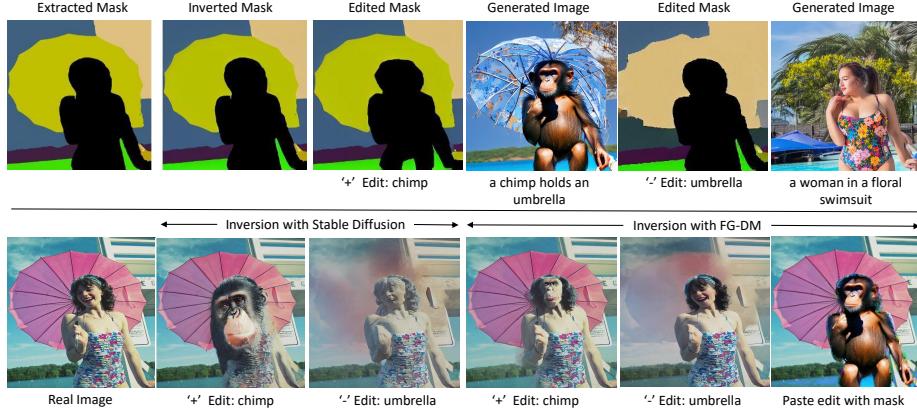
SBPC (Object Recall): Table 3, ablates the influence of noise seeds (batch size N) and number of sampling timesteps t on the prompt adherence (measured by object recall) of images synthesized by the FG-DM for ADE20K validation set prompts (8.5 objects per prompt on average). Avg.

8

Max Recall is the average, over the 2000 prompts, of the maximum per image recall in each batch. Similar definitions hold for Avg.Min and Avg. Median. The top part of the table (columns 2-4) shows that for $N = 1$ recall saturates at 69% for $t = 10$ timesteps. The bottom part shows that increasing N maintains the Avg. Median recall at this value but produces images with a significant variation of recall values. The Avg. Max recall is 6 points higher (75%) and fairly stable across configurations of N and t. The fifth column shows the number of images in a batch that satisfy the object recall thresholds of 0.5,0.75 and 0.9, averaged over 2000 prompts. While this number decreases for higher thresholds, a batch size of $N = 10$ can produce at least one image with even higher prompt compliance, on average.

The FG-DM is particularly well suited to implement SBPC, because recall can be computed as soon as segmentations are sampled. Since the segmentation factor can be run with a smaller number of timesteps and at lower resolution, this is much faster than sampling the images themselves. The image synthesis factor is run only once, for the segmentation mask of highest recall. On the contrary, an SD-based implementation of SBPC requires synthesizing N images and then feeding them to an external segmentation model (we used Segformer-B4 (50) in our experients) to compute recall. Table 4 compares the object recall of different sampling configurations of SD and FG-DM for ADE20K validation prompts. The objects in the groundtruth masks are considered to compute recall. In appendix section A.2.1, we show that the object classes can be automatically extracted from the prompt using an LLM. The top part of the table compares single run ($N = 1$) implementations of SBPC by SD and the FG-DM to the popular IBPC Attend & Excite (A-E) method (7). The table shows that the FG-DM images have **8%** and **4%** higher recall than those of SD and A-E respectively, even though SD has higher clip score. This illustrates the deficiencies of clip score to evaluate prompt adherence. The IBPC method underperforms the FG-DM by 4 points while drastically increasing inference time to 36 seconds per prompt. This is because A-E computation scales linearly with the number of objects and it fails in multiple object scenarios. While SBPC achieves good results for $N = 1$, the bottom half of the table shows that its true potential is unlocked by larger batch sizes. Both the SD and FG-DM implementations of SBPC achieved the much higher average recall of **75**%. However, for SD, the sampling of a batch of $N = 10$ high resolution images requires an amount of computation (23 seconds per prompt) prohibitive for most applications. The FG-DM is **4x (3x) faster** when using 10 (20) DDIM steps for the segmentation mask and 20 steps for image synthesis.

Besides object recall, the FG-DM can also improve the recall of semantic attributes. We illustrate this with an experiment in face synthesis for FG-DM trained from scratch on MM-CelebA-HQ(49). We compute the recall of semantic attributes such as bald, necklace, and pale skin etc. using the generated segmentation masks on validation prompts. Fig. 7 (left) compares the attribute recall of FG-DM to those of prior methods for $N = 1$. The FG-DM has average recall of **75**% over all semantic attributes, outperforming competing methods (32). Figure 7 (right) shows an histogram of the number of trials (N) required by the FG-DM to achieve 70% and 80% recall[1]. The FG-DM generates $\approx 90\%$ of its samples with the specified 70% recall in five trials.

Attention distillation loss: Table 2 ablates the adaptation of SD with and without attention distillation loss. The FG-DM with attention distillation improves the image quality by **1.29/1.3** points for segmentation/normal map conditions respectively. The effect is more pronounced when comparing the fidelity of the conditions which improves the quality by **5.6/2.8** points for segmentation/normal synthesis showing the effectiveness of the loss in adapting SD to different conditions. Appendix Figure 9 shows the qualitative comparison of ablating the attention distillation loss.

Real Image Editing with FG-DM: We show the results of editing of both real images and their segmentation masks with FG-DM. The top of Figure 8 refers to inversion of the segmentation mask. We use an off-the-shelf OpenSEED (57) model to extract the segmentation map of a real image (shown on the bottom left of the figure) and apply the FG-DM segmentation factor model for inversion and editing using LEDITS++ (4), a recent method for text based image editing. We apply LEDITS++ to the segmentation factor to 1) replace the mask of the woman by that of a chimp (third image of the top row) and 2) to delete the mask of the umbrella (fifth image). New images (fourth and sixth) are then generated by the image synthesis factor conditioned on the edited segmentation masks. We have found that the inversion and editing of segmentation masks is quite robust. The synthesized masks usually reflect the desired edits. However, because the final image synthesis is only conditioned on these masks, the synthesized image does not maintain the background of the original image. The

---

[1]10 runs includes samples that required 10 or more trials.

9

synthesized image is a replica of the original image at the semantic level (similar objects and layout) but not at the pixel level. From our experiments, this method has high robustness and quality for semantic-level editing.

We next investigated pixel level inversion and editing, which is harder. The bottom part of Figure 8 shows the comparison of LEDITS++ editing with inversion by SD and by the image synthesis factor of the FG-DM. For the latter, we apply inversion to the ControlNet image generation factor using the real image and the segmentation mask extracted from it. Then we perform the LEDITS++ edit using the edited mask from the top part of Figure 8 (inverted with the FG-DM segmentation factor) to produce the edited image as shown in columns 4 and 5. This pixel-level inversion and editing tends to maintain the background of the original image but is much less robust than mask-level editing in terms of editing quality. This can be seen from the images in columns 2 and 3, which show the inversion using SD, which fails to produce a realistic chimp and turns the woman into a stone sculpture. The FG-DM produces much more meaningful edits, as shown in columns 4 and 5. The last column of the bottom part of the Figure 8 shows an added advantage of FG-DM where the chimp generated in the top portion can be pasted to the original image due to availability of the segmentation mask. In this example the pasting is rough around the object edges since we have made no attempts to beautify it. It can be improved by denoising the generated image with one forward pass of SD at a higher timestep.

Figure 8: **Top:** Inverting segmentation masks with FG-DM segmentation factor using the LEDITS++ method. Edits to replace the woman by a chimp or eliminate the umbrella. The FG-DM enbables text-based edits to modify or delete objects in a given mask. The image generated with the edited mask as condition is shown to the right of each edited masks. **Bottom:** Original image, LEDITS++ edited image for stable diffusion and for the image synthesis factor of the FG-DM. Please Zoom in for details.

5   Limitations, Future Work and Conclusion

Although, the FG-DM uses low resolution synthesis for the conditions, runtime increases for chains with more than two factors. Since, the attention maps generated per factor must be consistent according to the joint model, sharing them across different factors and timesteps is a promising direction for further reducing the runtime. Furthermore, while the FG-DM allows easier control over generated images for operations like deleting, moving, or flipping objects, fine-grained manipulations (e.g. changing the branches of a tree) can still require considerable user effort. Automating the pipeline by using an LLM or methods like Instructpix2pix (5) to instruct the edits of the synthesized conditions is another interesting research direction. See A.3 for a discussion on broader impact.

In this work, we proposed the FG-DM framework for efficiently adapting SD for improved prompt compliance and controllable image synthesis. We showed that an FG-DM trained with relatively small datasets generalizes to prompts beyond these datasets, supports fine-grained image editing, enables improved prompt compliance by SBPC, allows adding new conditions without having to retrain all existing ones, and supports data augmentation for training downstream models. It was also shown that the FG-DM enables faster and creative image synthesis, which can be tedious or impossible with existing conditional image synthesis models. Due to this, we believe that the FG-DM is a highly flexible, modular and useful framework for various image synthesis applications.

10

Acknowledgements

This work was partially funded by the NSF award IIS-2303153. We also acknowledge and thank the use of the Nautilus platform for some of the experiments discussed above.

References

[1] Avrahami, O., Hayes, T., Gafni, O., Gupta, S., Taigman, Y., Parikh, D., Lischinski, D., Fried, O., Yin, X.: Spatext: Spatio-textual representation for controllable image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 18370–18380 (June 2023)

[2] Azadi, S., Tschannen, M., Tzeng, E., Gelly, S., Darrell, T., Lucic, M.: Semantic bottleneck scene generation. arXiv preprint arXiv:1911.11357 (2019)

[3] Birkl, R., Wofk, D., Müller, M.: Midas v3.1 – a model zoo for robust monocular relative depth estimation. arXiv preprint arXiv:2307.14460 (2023)

[4] Brack, M., Friedrich, F., Kornmeier, K., Tsaban, L., Schramowski, P., Kersting, K., Passos, A.: Ledits++: Limitless image editing using text-to-image models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)

[5] Brooks, T., Holynski, A., Efros, A.A.: Instructpix2pix: Learning to follow image editing instructions. In: CVPR (2023)

[6] Casanova, A., Careil, M., Romero-Soriano, A., Pal, C.J., Verbeek, J., Drozdzal, M.: Controllable image generation via collage representations. arXiv preprint arXiv:12304.13722 (2023)

[7] Chefer, H., Alaluf, Y., Vinker, Y., Wolf, L., Cohen-Or, D.: Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. SIGGRAPH (2023)

[8] Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence **40**(4), 834–848 (2017)

[9] Cheng, J., Liang, X., Shi, X., He, T., Xiao, T., Li, M.: Layoutdiffuse: Adapting foundational diffusion models for layout-to-image generation. arXiv preprint arXiv:2302.08908 (2023)

[10] Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. Advances in Neural Information Processing Systems **34**, 8780–8794 (2021)

[11] Forney, G.D.: Codes on graphs: Normal realizations. IEEE Transactions on Information Theory **47**(2), 520–548 (2001)

[12] Gafni, O., Polyak, A., Ashual, O., Sheynin, S., Parikh, D., Taigman, Y.: Make-a-scene: Scene-based text-to-image generation with human priors. In: Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XV. p. 89–106. Springer-Verlag, Berlin, Heidelberg (2022), `https://doi.org/10.1007/978-3-031-19784-0_6`

[13] Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., Cohen-Or, D.: Prompt-to-prompt image editing with cross attention control. arXiv preprint arXiv:2208.01626 (2022)

[14] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017), `https://proceedings.neurips.cc/paper_files/paper/2017/file/8a1d694707eb0fefe65871369074926d-Paper.pdf`

[15] Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 6840–6851. Curran Associates, Inc. (2020), `https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf`

[16] Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems **33**, 6840–6851 (2020)

[17] Ho, J., Salimans, T.: Classifier-free diffusion guidance (2022)

11

[18] Jin, S., Xu, L., Xu, J., Wang, C., Liu, W., Qian, C., Ouyang, W., Luo, P.: Whole-body human pose estimation in the wild. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)

[19] Ju, X., Zeng, A., Zhao, C., Wang, J., Zhang, L., Xu, Q.: HumanSD: A native skeleton-guided diffusion model for human image generation. arXiv preprint arXiv:1904.06539 (2023)

[20] Kynkäänniemi, T., Karras, T., Laine, S., Lehtinen, J., Aila, T.: Improved precision and recall metric for assessing generative models. CoRR **abs/1904.06991** (2019)

[21] Le Moing, G., Vu, T.H., Jain, H., Pérez, P., Cord, M.: Semantic palette: Guiding scene generation with class proportions. In: CVPR (2021)

[22] Lee, C.H., Liu, Z., Wu, L., Luo, P.: Maskgan: Towards diverse and interactive facial image manipulation (2020)

[23] Lin, T., Maire, M., Belongie, S.J., Bourdev, L.D., Girshick, R.B., Hays, J., Perona, P., Ramanan, D., Doll'a r, P., Zitnick, C.L.: Microsoft COCO: common objects in context. CoRR **abs/1405.0312** (2014), `http://arxiv.org/abs/1405.0312`

[24] Liu, X., Ren, J., Siarohin, A., Skorokhodov, I., Li, Y., Lin, D., Liu, X., Liu, Z., Tulyakov, S.: Hyperhuman: Hyper-realistic human generation with latent structural diffusion. arXiv preprint arXiv:2310.08579 (2023)

[25] Luo, X., Goebel, M., Barshan, E., Yang, F.: Leca: A learned approach for efficient cover-agnostic watermarking (2022)

[26] Mirjalili, S.: if-ai-image-generators-are-so-smart-why-do-they-struggle-to-write-and-count (2023), `https://theconversation.com/if-ai-image-generators-are-so-smart-why-do-they-struggle-to-write-and-count-208`

[27] Mizrahi, D., Bachmann, R., Kar, O.F., Yeo, T., Gao, M., Dehghan, A., Zamir, A.: 4m: Massively multimodal masked modeling. In: Thirty-seventh Conference on Neural Information Processing Systems (2023), `https://openreview.net/forum?id=TegmlsD8oQ`

[28] Mizrahi, D., Bachmann, R., Kar, O.F., Yeo, T., Gao, M., Dehghan, A., Zamir, A.: 4M: Massively multimodal masked modeling. In: Advances in Neural Information Processing Systems (2023)

[29] Mou, C., Wang, X., Xie, L., Wu, Y., Zhang, J., Qi, Z., Shan, Y., Qie, X.: T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. arXiv preprint arXiv:2302.08453 (2023)

[30] Nichol, A.Q., Dhariwal, P.: Improved denoising diffusion probabilistic models. In: International Conference on Machine Learning. pp. 8162–8171. PMLR (2021)

[31] Obukhov, A., Seitzer, M., Wu, P.W., Zhydenko, S., Kyl, J., Lin, E.Y.J.: High-fidelity performance metrics for generative models in pytorch (2020). https://doi.org/10.5281/zenodo.4957738, `https://github.com/toshas/torch-fidelity`, version: 0.3.0, DOI: 10.5281/zenodo.4957738

[32] Park, M., Yun, J., Choi, S., Choo, J.: Learning to generate semantic layouts for higher text-image correspondence in text-to-image synthesis. ICCV (2023)

[33] Peng, Y., Zhao, C., Xie, H., Fukusato, T., Miyata, K.: Difffacesketch: High-fidelity face image synthesis with sketch-guided latent diffusion model. arXiv preprint arXiv:2302.06908 (2023)

[34] Phung, Q., Ge, S., Huang, J.B.: Grounded text-to-image synthesis with attention refocusing. In: CVPR (2024)

[35] Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. arXiv (2022)

[36] Radford, A., Kim, J., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763 (2021)

[37] Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. ArXiv **abs/2204.06125** (2022), `https://api.semanticscholar.org/CorpusID:248097655`

[38] Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-shot text-to-image generation. In: International Conference on Machine Learning. pp. 8821–8831. PMLR (2021)

12

[39] Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., Koltun, V.: Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(3) (2022)

[40] Rassin, R., Hirsch, E., Glickman, D., Ravfogel, S., Goldberg, Y., Chechik, G.: Linguistic binding in diffusion models: Enhancing attribute correspondence through attention map alignment. Advances in Neural Information Processing Systems **36** (2024)

[41] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10684–10695 (2022)

[42] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. pp. 234–241. Springer (2015)

[43] Sagonas, C., Tzimiropoulos, G., Zafeiriou, S., Pantic, M.: 300 faces in-the-wild challenge: The first facial landmark localization challenge. In: 2013 IEEE International Conference on Computer Vision Workshops. pp. 397–403 (2013). https://doi.org/10.1109/ICCVW.2013.59

[44] Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: Bach, F., Blei, D. (eds.) Proceedings of the 32nd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 37, pp. 2256–2265. PMLR, Lille, France (07–09 Jul 2015)

[45] Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. In: International Conference on Learning Representations (2021), `https://openreview.net/forum?id=St1giarCHLP`

[46] Truly, A.: 5-things-ai-image-generators-still-struggle-with (2023), `https://www.digitaltrends.com/computing/5-things-ai-image-generators-still-struggle-with/`

[47] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)

[48] Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., Xiao, B.: Deep high-resolution representation learning for visual recognition. TPAMI (2019)

[49] Xia, W., Yang, Y., Xue, J.H., Wu, B.: Tedigan: Text-guided diverse face image generation and manipulation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2021)

[50] Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. In: Neural Information Processing Systems (NeurIPS) (2021)

[51] "Xie, S., Tu, Z.: Holistically-nested edge detection. In: Proceedings of IEEE International Conference on Computer Vision (2015)

[52] Yang, B., Gu, S., Zhang, B., Zhang, T., Chen, X., Sun, X., Chen, D., Wen, F.: Paint by example: Exemplar-based image editing with diffusion models. arXiv preprint arXiv:2211.13227 (2022)

[53] Yang, L., Huang, Z., Song, Y., Hong, S., Li, G., Zhang, W., Cui, B., Ghanem, B., Yang, M.H.: Diffusion-based scene graph to image generation with masked contrastive pre-training. arXiv preprint arXiv:2211.11138 (2022)

[54] Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: Bisenet: Bilateral segmentation network for real-time semantic segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV) (September 2018)

[55] Yu, F., Koltun, V., Funkhouser, T.: Dilated residual networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)

[56] Yu, J., Xu, Y., Koh, J.Y., Luong, T., Baid, G., Wang, Z., Vasudevan, V., Ku, A., Yang, Y., Ayan, B.K., Hutchinson, B., Han, W., Parekh, Z., Li, X., Zhang, H., Baldridge, J., Wu, Y.: Scaling autoregressive models for content-rich text-to-image generation. Transactions on Machine Learning Research (2022), `https://openreview.net/forum?id=AFDcYJKhND`, featured Certification

[57] Zhang, H., Li, F., Zou, X., Liu, S., Li, C., Gao, J., Yang, J., Zhang, L.: A simple framework for open-vocabulary segmentation and detection. arXiv preprint arXiv:2303.08131 (2023)

[58] Zhang, L., Agrawala, M.: Adding conditional control to text-to-image diffusion models. ICCV (2023)

13

[59] Zhang, R., Isola, P., Efros, A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. CVPR (01 2018)

[60] Zhao, S., Chen, D., Chen, Y.C., Bao, J., Hao, S., Yuan, L., Wong, K.Y.K.: Uni-controlnet: All-in-one control to text-to-image diffusion models. Advances in Neural Information Processing Systems (2023)

[61] Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017)

14

A Appendix

Figure 9: Comparison of FG-DM with attention distill loss (bottom) against FG-DM without attention distill loss (center) and the recent 4M-XL model (top) for the prompts shown at the top of the figure. The two versions of the FG-DM use the same seed. Both versions of the FG-DM produce images of higher quality than 4M-XL. Attention distillation helps improve the quality of the generated segmentations. For example, the model without distillation has inaccurate masks/missing cart/less realistic zebras from left to right.

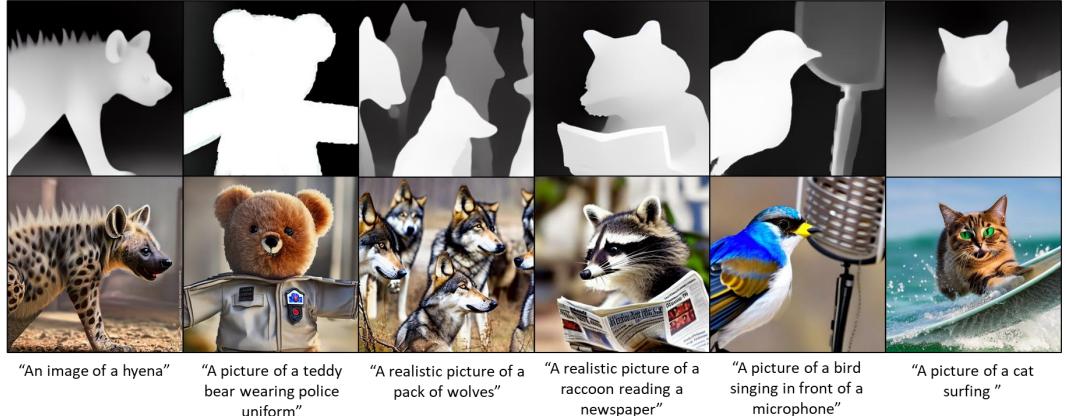



Figure 10: **Qualitative results** of Depth map/Image pairs synthesized by FG-DM.

A.1 Experimental Results

A.1.1 FG-DMs Adapted from Stable Diffusion (SD)

We start by the discussion of additional results for FG-DMs obtained by adaptation of Stable Diffusion model, as shown in Figure 4 of the paper.

Qualitative comparison of attention distillation loss

Figure 9 shows some qualitative results of the ablation for the impact of the attention distillation loss. There is a clear qualitative benefit in introducing this loss. Without it, the model generates less accurate masks, leading to an unrealistic pizza making depiction/ cart-person relationship/ zebra pair from left to right. This confirms the qualitative ablation showing the benefits of the attention distillation loss in Table 2 but provides a stronger illustration of the advantages of the loss, which

15

"A picture of a polar bear" | "A statue of virgin mary holding an orb of light" | "A picture of a blue elephant" | "A kitchen with a counter and a table with chairs." | "Closeup of a plate with chicken, mushrooms and broccoli." | "A large jetliner on an airport runway."

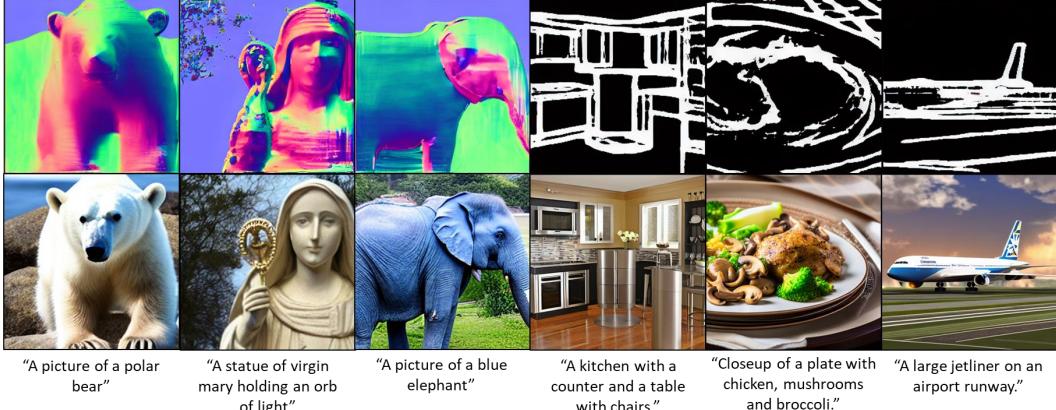



Figure 11: **Qualitative results** of Normal map/image and Sketch map/image pairs synthesized by FG-DM. FG-DM generalizes well across conditions and is able to generate condition-image pairs that are not seen during training.

"a woman standing in a kitchen" | "a person and a white chimp posing" | "A plate has a waffle, some fruit and ice cream on it" | "A boat on the water with a lighthouse in background" | "A bride and a groom at the backdrop of mountains" | "A small bathroom with a sink and toilet"

Figure 12: **Qualitative results** of Segmentation map/image pairs synthesized by FG-DM. The second column shows the benefit of explainability with FG-DM which allows verifying intermediate conditions to understand the hallucinations (mixup of a chimp and a person) of SD which are opaque otherwise. Here, FG-DM correctly generates the chimp and the person mask with different colors while ControlNet confuses between the two showing that the ControlNet needs to be corrected.

tends to produce more "plausible" scenes. Such plausibility is difficult to measure with qualitative metrics. For example, the CLIP score is not sensitive to the fact that the cart and the person are not interacting in a normal way, or that the pizza making activity is unrealistic.

We also compared with the recent 4M model (28), an autoregressive model trained from scratch on both discriminative and generative tasks. In this case, we use the largest model (4X-ML) released by the authors. Figure 9 shows a qualitative comparison between FG-DM and 4M. It can be seen that 4M generates images of weaker quality (distorted hands, missing person's head, deformed zebra bodies) as compared to FG-DM with/without the attention distillation loss.

Additional Qualitative Results for Segmentation, Depth, Normal and Sketch conditions

Figure 10 shows qualitative results of synthesized depth maps and images for the creative prompts shown on top/bottom of each image. The FG-DM framework is able to generate high quality images and normal maps for prompts that are not seen in the training setleveraging the generalization of SD.

Figure 11 (first three columns) shows qualitative results of synthesized normal maps and images for the creative prompts shown on top/bottom of each image. Figure 11 (last three columns) shows qualitative results of synthesized sketch maps and images for the creative prompts shown on top/bottom of each image.

16

	Inputs	MAS	SpaText	FG-DM	FG-DM

a mug and a white plate with cookies on a table

A black elephant near a lake

A bathroom with an artificial light, a mirror, a white sink and a vase with red flowers

Figure 13: **Qualitative comparison of FG-DM with prior works** such as Make-a-Scene and SpaText for the prompts shown on the left. Note that FG-DM generates both the map and the image while for MAS and SpaText, the condition was manually sketched and fed to the model.

Figure 12 shows qualitative results of synthesized segmentation maps and images for the creative prompts shown on top/bottom of each image. As shown in the main paper, the FG-DM is able to synthesize segmentation maps for object classes beyond the training set and the semantic maps are colored with unique colors, allowing the easy extraction of both object masks and class labels. This shows the potential of the FG-DM for open-set segmentation, e.g the synthesis of training sets to generate training data for segmentation models. Further, a number of interesting generalization properties emerge. Although the FG-DM is only trained to associate persons with black semantic maps segments, it also assigns the chimp of Figure 1, a class that it was not trained on, to that color. This shows that the FG-DM can integrate the prior knowledge by SD that "chimps and persons are similar" into the segmentation task, which receives supervision from COCO alone. Conversely, the similarity between chimps and people might induce SD to synthesize a chimp in response to a prompt for people, or vice-versa. This is shown in the bottom left of Figure 12 where the FG-DM correctly synthesizes different colors for the chimp and the person, but the ControlNet fails. While the black box nature of SD makes these errors opaque, the FG-DM allows inspection of the intermediate conditions to understand these hallucinations and make corrections accordingly. For example, in the above example ControlNet has to be corrected by either finetuning or using inference optimization methods like A-E (7). This illustrates its benefits in terms of explainability.

Comparison with prior works

Figure 13 shows the qualitative comparison of FG-DM with prior works such as Make-a-Scene (12) or SpaText (1) in addition to 4M model compared in Figure 9. Note that FG-DM generates both the segmentation and the image while for the other methods, it is manually sketched and fed to them. It is seen that FG-DM generates high quality images that adhere well to the prompts as compared to the prior works.

Comparison of generated conditions by FG-DM to conditions recovered by off-the-shelf models

Figure 15 shows a qualitative comparison of the conditions synthesized by FG-DMs to those recovered from the synthesized image using off-the-shelf pretrained models for segmentation and depth estimation. The qualitative results corroborate with the user study as the generated conditions are better than extracted ones for depth while they are similar for segmentation conditions.

Synthesis of Conditioning Variables with SD Autoencoder

17

"An image of a small rhino"   "An image of a giant squirrel"   "A man eating a hotdog"   "A picture of a volcanic eruption"   "A picture of a sungod"

"A penguin riding a bicycle"   "A man with a hat and glasses"   "A bird with headphones singing in front of a microphone"   "An older woman sitting near a window of a train."   "A room with a fireplace and a television"

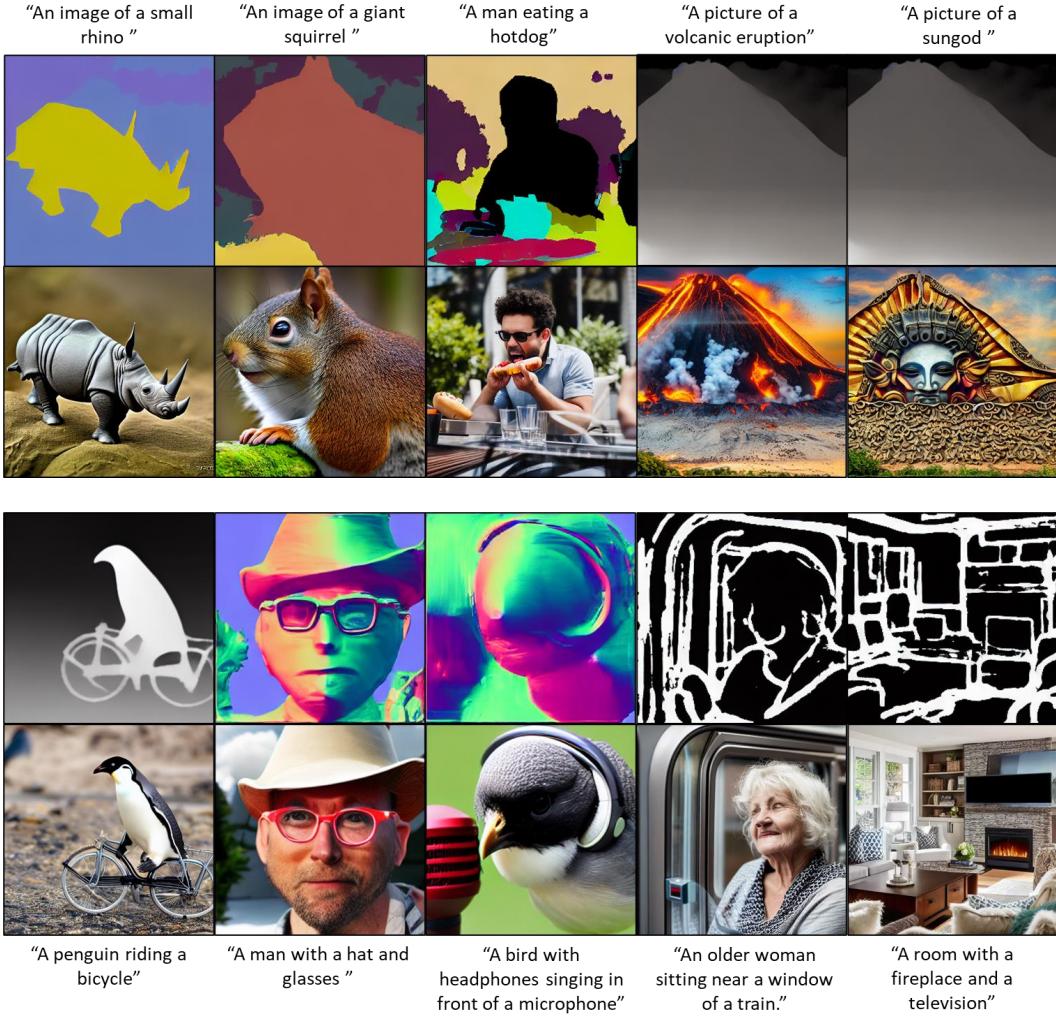
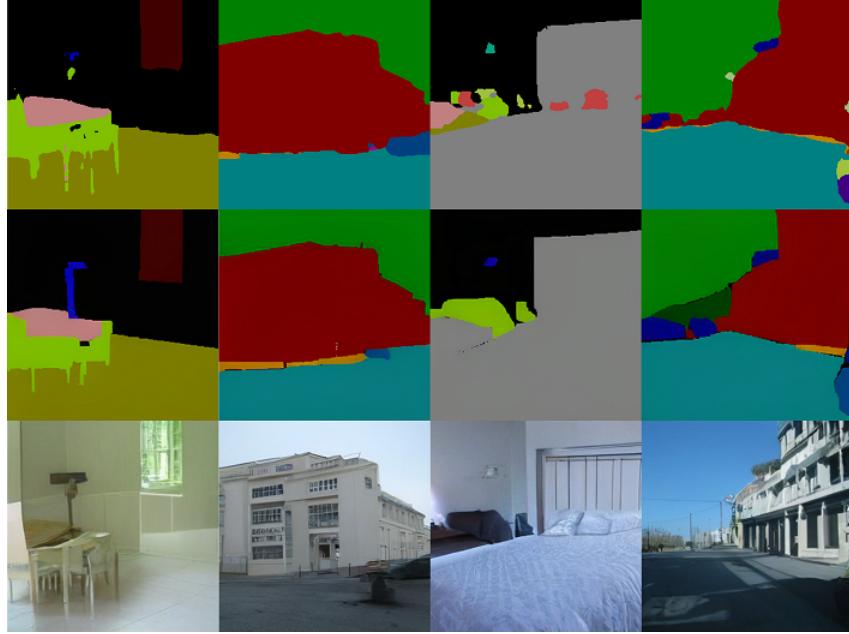



Figure 14: **Zoomed version of Figure 5** Qualitative results of FG-DM to synthesize segmentation, depth, normal and sketch maps and their corresponding images.

High-quality image synthesizes requires DMs trained from large datasets. A common solution is to adopt the LDM (41) architecture, where an encoder ($\mathcal{E}$)-decoder ($\mathcal{D}$) pair is used to map images into a lower dimensional latent space. Using a pre-trained $\mathcal{E}$-$\mathcal{D}$ pair, e.g. from the SD model, guarantees that latent codes map into high quality images, making it possible to train high-fidelity DMs with relatively small datasets. However, it is unclear that this approach will work for the synthesis of conditioning variables, such as segmentation maps, which SD is not trained on. For example, (21; 32) explicitly address training DMs to produce the discrete outputs required by many conditioning variables. Somewhat surprisingly, our experiments show that off-the-shelf foundation DMs are quite effective at synthesizing visual conditioning variables. All our results use the following procedure: 1) visual conditioning variables are converted to 3 channel inputs. For discrete variables, a different color is simply assigned to each variable value (e.g. each semantic label of a segmentation map). 2) All VC-DMs in the FG-DM re-use the pre-trained encoder-decoder pair of SD (41), as illustrated in Figure 4. 3) At the decoder output, discrete values are recovered with a hashmap that matches image colors to their color prototypes. This is done by simple thresholding, with a margin threshold empirically set to 28 for each pixel. To test this procedure, we measured the mean squared pixel reconstruction error of the segmentation maps from 2,000 validation images of the ADE20K dataset. This was 0.0053 (normalized) or 1.34 pixel error showing that the pretrained SD autoencoder is highly effective in reconstructing discrete maps. Figure 16 compares the auto-encoder reconstructed maps against the groundtruth maps for different conditioning variables such as segmentation, depth, sketch and normal. This shows that simply representing the conditioning variables as 3-channel

18

inputs allows the faithful reconstruction of all conditions. Further, we also trained a lightweight segmentation model to recover the labels from the RGB semantic maps but observed no improvement over this simple heuristic.

(a) Segmentation maps. (Middle: Generated, Top: Extracted)

(b) Depth maps. (Middle: Generated, Top: Extracted)

Figure 15: Qualitative comparison of generated conditions for FG-DM vs extracted conditions using SD (Stable Diffusion) + CEM (Condition Extraction Model) for segmentation and depth maps. The generated conditions for depth maps are superior to the extracted ones.

Computation-Performance Tradeoff:

The image generation chain (final factor) of FG-DM has the same computation as existing VC-DMs since it re-uses existing conditional models such as ControlNet. The only additional computation comes from the condition synthesis factor and a faithful comparison of complexity must include the condition synthesis step. For existing DMs, generating/editing conditions for segmentation

19

(a) Segmentation maps. (Left: Groundtruth, Right: Reconstructed)

(b) Depth maps. (Left: Groundtruth, Right: Reconstructed)

(c) Sketch maps. (Left: Groundtruth, Right: Reconstructed)

(d) Normal maps. (Left: Groundtruth, Right: Reconstructed)

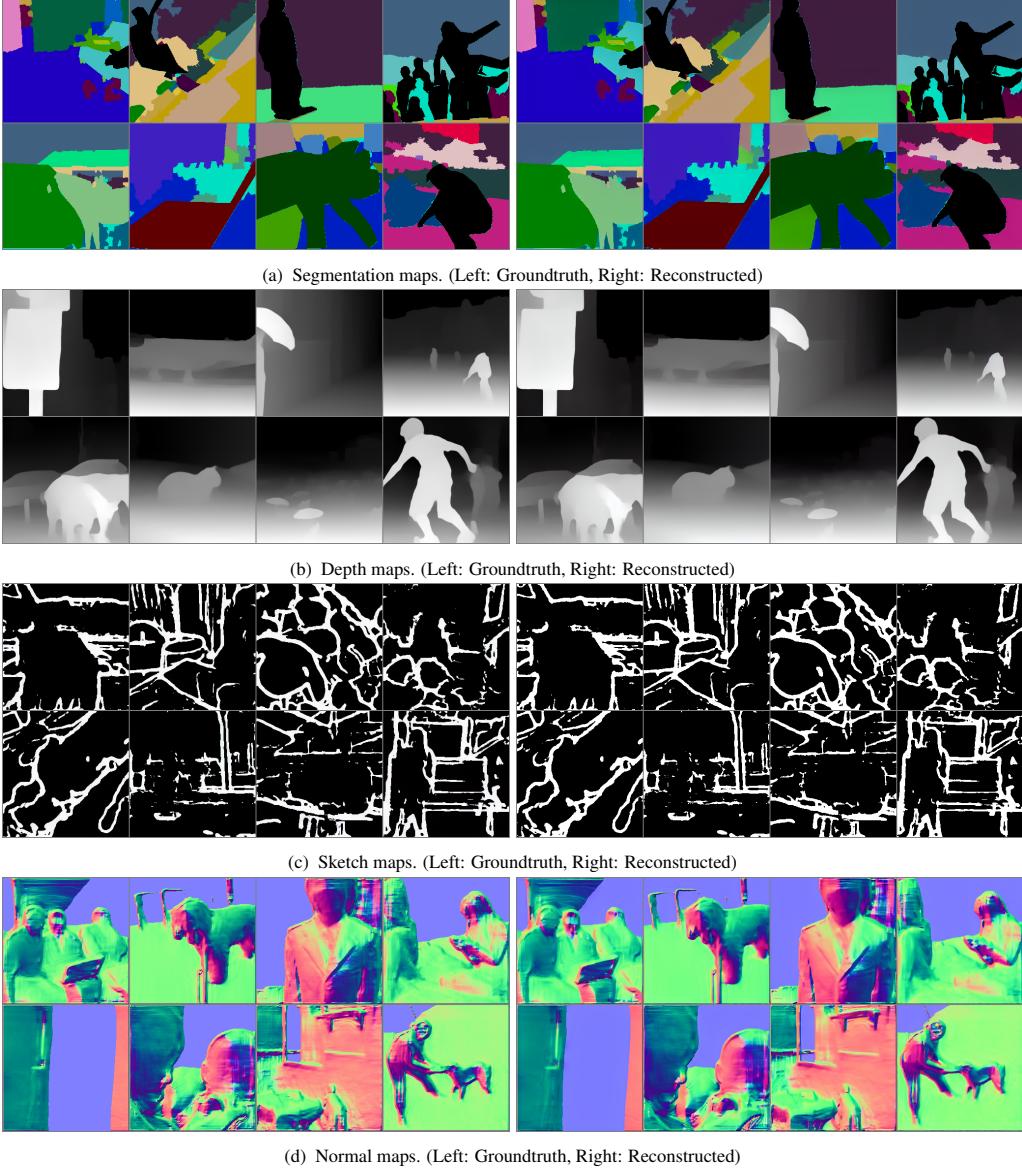



Figure 16: Visualization of groundtruth (left) and reconstructed (right) maps by applying the pretrained stable diffusion autoencoder to segmentation, depth, sketch and normal maps.

Table 6: Comparison of segmentation mask quality and throughput of FG-DM trained on ADE20K dataset against state-of-the-art conventional segmentation models of similar size. The FG-DM samples higher quality masks with only 10 DDIM steps, comparable to the throughput of Segformer-B5 but with superior quality as shown in the FID, Precision and Recall metrics.

Model (ADE20K)	Params (M)	FID↓	LPIPS↑	Precision↑	Recall↑	T (imgs/s)↑
SegFormer-B5 (50)	85	112.6	0.781	0.61	0.04	**6.4**
FG-DM-Seg (10 DDIM steps)	53	**86.1**	**0.788**	**0.72**	**0.04**	**6.4**
FG-DM-Seg (20 DDIM steps)	53	**84.0**	0.776	**0.72**	**0.05**	3.7
FG-DM-Seg (200 DDIM steps)	53	**83.6**	0.768	**0.73**	**0.06**	0.4

masks requires the following steps: (1) Generate image with DM. (2) Segment with off-the-shelf segmentation model. (3) Edit the segmentation. (4) Generate the image conditional on the manipulated segmentation. The FG-DM eliminates step 1. replaces step 2. with the segmentation synthesis step. This is much more efficient than running an image generation DM and a segmentation model as illustrated in Table 4 since FG-DM samples segmentations at a lower resolution. For example, a 20-step image generation, FG-DM (with ControlNet as the final factor) takes only 4.5s **(1.7x speedup)** as compared to 7.5s for the standard pipeline (image generation with SD, Segmentation with SegFormer (50)(CEM), and conditional image generation with ControlNet).

20

Table 7: **Ablation study on Image Synthesis** by FG-DM trained separately and jointly with segmentation factor. FG-DM results presented as Images/Semantic maps.

Model	#P (M)	MM-CelebA				CityScapes				ADE-20K				COCO-Stuff			
		FID ↓	LPIPS ↑	Pr↑	Re↑	FID ↓	LPIPS ↑	Pr↑	Re↑	FID ↓	LPIPS ↑	Pr↑	Re ↑	FID ↓	LPIPS ↑	Pr ↑	Re ↑
FG-DM (Separate training)	140	23.2/20.8	0.57/0.54	**0.83**	0.29	54.7/61.8	0.56/0.54	0.66	0.17	34.0/83.9	0.79/0.77	0.71	0.25	35.3/**40.6**	0.83/0.8	**0.71**	0.33
FG-DM (Joint training)	140	**21.3/20.3**	**0.58/0.54**	0.81	0.34	**47.6/61.8**	**0.59/0.57**	**0.69**	0.31	**29.6/83.6**	0.79/0.77	0.72	0.34	**33.1**/57.4	**0.83/0.8**	0.69	0.43

Table 8: Ablation study on data augmentation with synthetic data generated by the FG-DM for facial part segmentation on MM-CelebA-HQ(22) dataset.

Data	#Samples	mIoU↑	F.W. mIoU↑	F1-score↑
Orig	24993	51.1	85.6	61.5
Orig+Syn	+1000	**55.1**	**87.9**	**65.5**
Orig+Syn	+2000	55.0	87.1	65.0

Table 9: Ablation study on data augmentation with synthetic data generated by the FG-DM for face landmark estimation on 300W(43) dataset.

NME	#Samples	Common↓	Full↓	Challenge↓
Orig	3000	3.21	3.64	5.81
Orig+Syn	+1000	**3.12**	**3.54**	**5.80**
Orig+Syn	+2000	3.18	3.61	5.81

Next, we evaluate the generated mask quality from FG-DM. Table 6 compares the segmentation mask quality vs. throughput, for FG-DM and SegFormer (CEM) on the ADE20K dataset. In these experiments, SegFormer are applied to the validation dataset images and FG-DM masks are obtained with the validation prompts. Throughput is calculated on a single NVIDIA-TitanXp GPU with batch size 1 averaged over 2,000 runs. Performance is reported in terms of FID, LPIPS, Precision, and Recall metrics of the masks. For the FG-DM, results are presented at different sampling steps.

With only 10 timesteps, the FG-DM produces segmentations of quality superior to those of the CEM segmentation model. This is achieved with a throughput comparable to that of the bigger SegFormer-B5 model. Beyond 10 steps there are negligible improvements in all performance metrics, showing that conditions like segmentation masks can be generated much faster than natural images, due to their lower frequency content. The FG-DM merges the capability to generate segmentation masks of superior quality compared to current segmentation models, *crucial for creative tasks with unseen images*, with a throughput comparable to the latter, which can only segment existing images.

A.1.2   Models trained from scratch: Ablation Studies

We next discuss some results for FG-DMs trained from scratch where the adapter in Fig. 4 is removed and the intermediate conditions are concatenated to be fed to the subsequent factors.

Joint Synthesis: Table 7 compares end-to-end training of FG-DM to separate training of VC-DM factors. Jointly training the FG-DM improves the image quality (lower FID) and diversity (higher LPIPS) on all four datasets.

Table 10 shows the comparison of FG-DM with joint modeling by concatenation, where conditioning variable(s) are concatenated in the latent space and denoised jointly with the image using a single DM. The table clearly shows that FG-DM outperforms the concatenation approach by **8** points on the FID metric despite being smaller in size. Note that joint denoising by concatenation requires a larger model and forfeits many advantages of the FG-DM, such as higher object recall, image editing, and computational efficiency.

Order of conditions: Table 11 ablates the order of the conditioning variables for the model with semantic (S) and pose (P) condition on CelebA-HQ and COCO datasets which shows that the order of the chain affects final image synthesis quality (see FID). This primarily stems from the misalignment of the pose map with semantic map and image, where the generated pose maps are inferior for the alternate order (pose-semantic-image). It shows that the unconditional generation of sparse pose maps is much more difficult than segmentations, due to the incomplete poses (occlusion) and missing context in the pose images. Further, Table 11 shows that the extra pose conditioning improves the image synthesis quality, reducing FID by **3.7/2.33** points on CelebA-HQ/COCO datasets respectively. This shows that adding more conditions is helpful for improving the image quality.

Data Augmentation

An additional benefit of the FG-DM is that it can be used as a data augmentation technique, synthesizing data to train models for segmentation, pose estimation, etc. By sampling data from the FG-DM, it

21

Table 10: Comparison of FG-DM conditioning vs concatenation approach for joint synthesis on CelebA-HQ. U-LDM reported for reference.

Model	#P (M)	FID ↓	LPIPS ↑
U-LDM (41)	87	24.3	0.586
Joint (Concat)	248	29.4	**0.618**
FG-DM (Ours)	**140**	**21.3**	0.578

Table 11: Ablation on the order of generated semantic map (S) and pose (P) conditions on CelebA-HQ (Top) and COCO (Bottom). I - Image.

Model	FID ↓	LPIPS ↑	P ↑	R ↑
(P→S→I)	23.43	**0.616**	0.616	**0.466**
(S→P→I)	**17.61**	0.594	**0.754**	0.403
(P→S→I)	31.43	0.855	0.547	**0.343**
(S→P→I)	**30.77**	**0.857**	**0.564**	0.322

Table 12: **Ablation study on Image Synthesis** by FG-DM for sequential and joint inference. For the same segmentation maps, joint inference is superior to sequential inference showing the benefit of joint modeling.

Model	#P (M)	MM-CelebA FID ↓	LPIPS ↑	Pr↑	Re↑	Cityscapes FID ↓	LPIPS ↑	Pr↑	Re↑	ADE-20K FID ↓	LPIPS ↑	Pr ↑	Re ↑	COCO FID ↓	LPIPS ↑	Pr ↑	Re ↑
FG-DM (Sequential Inference)	140	34.5/20.3	0.56/0.54	0.79	0.20	57.2/61.8	0.59/0.57	0.44	0.11	31.1/83.9	0.78/0.77	0.71	0.32	35.3/57.4	0.82/0.8	0.67	0.41
FG-DM (Joint Inference)	140	21.3/20.3	0.58/0.54	0.81	0.34	47.6/61.8	0.59/0.57	0.69	0.31	29.6/83.6	0.79/0.77	0.72	0.34	33.1/57.4	0.83/0.8	0.69	0.43

is possible to produce labeled datasets of virtually unlimited size. These could be, in principle, useful to train downstream models for various vision applications. To investigate this, we start by training both the FG-DM and the downstream model on a labeled dataset A. We then use the FG-DM to synthesize an additional dataset B of images and labels and retrain the downstream model on $A \cup B$. We finally compare the performance of the two downstream models. We performed this experiment for two downstream tasks: *part segmentation* with the BiSeNet(54) network on MM-CelebAA-HQ, and *pose estimation* with the HRNetV2-W18 (48) network on the 300W (43) dataset. In all cases, the downstream model is initialized with ImageNet pre-trained weights. Tables 8 and 9 show that the addition of synthetic data always improves downstream model performance. The gains are larger for the more challenging segmentation task, where 1000 samples of synthetic data significantly improve the baseline mIOU and F1-score by **4%**. For pose estimation, the addition of synthetic data reduces the already low normalized mean error (NME) of keypoint location by an additional 0.1%. Performance saturates or decreases slightly when the number of synthetic images is increased to 2000 as compared to using only 1000 images. We conjecture that this occurs from the inclusion of redundant or noisy images in the enlarged dataset, which are less likely to contribute meaningfully to the model's training. Nevertheless, the performance still shows improvement compared to the baseline model that did not use any synthetic data augmentation. These results suggest that the FG-DM can be used for data augmentation, enabling significant performance gains by automated data augmentation.

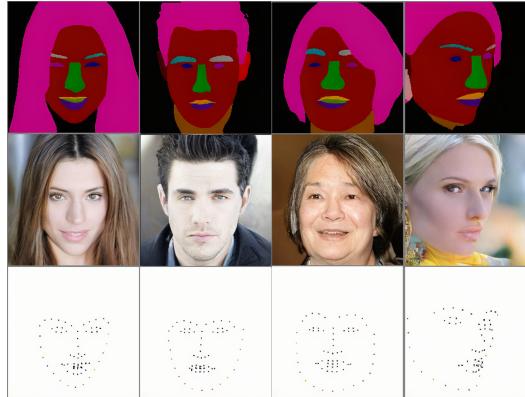



Figure 17: **Unconditional generated semantic, pose masks, and corresponding images using an FG-DM trained from scratch on MM-CelebA-HQ (256 x 256).**

Sequential vs Joint Inference Table 12 shows the comparison of two modes of operation of FG-DM during inference. Joint inference is the standard mode of operation that samples the condition(s) and image jointly at each timestep (equations (2)-(4) in the paper). Sequential inference is the alternative mode of operation where the condition synthesis chain is run fully before feeding it to the conditional image generation factor. It is observed that the standard mode of joint inference sampling produces images of higher quality (lower FID) than the alternative mode.

Qualitative results: Unconditional and Conditional Image Generation

22

(a) MM-CelebA-HQ 256 x 256 samples

(b) ADE-20K 256 x 256 samples

(c) Cityscapes 256 x 512 samples

(d) COCO 256 x 256 samples

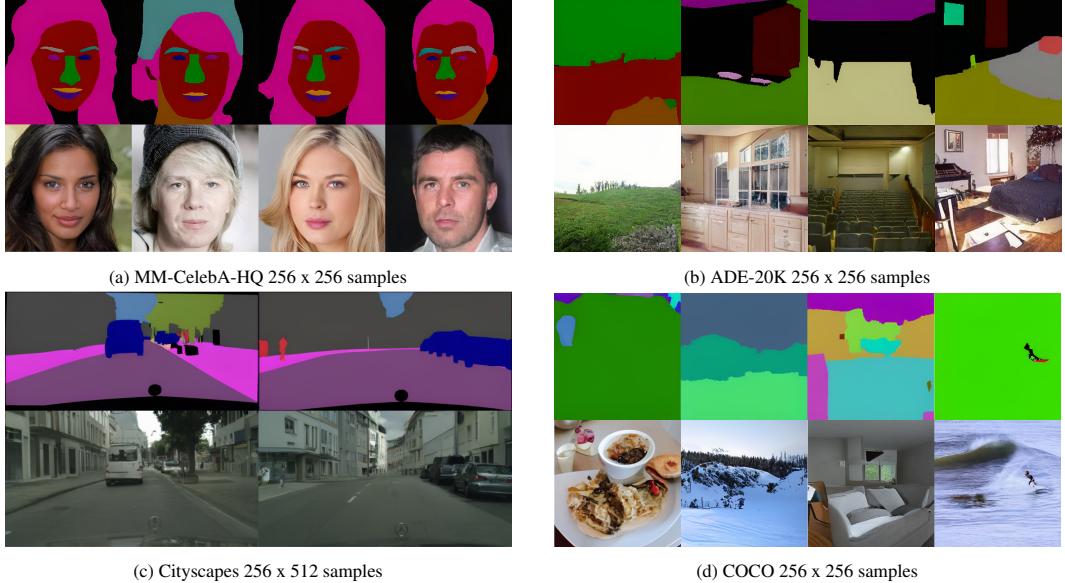



Figure 18: **Segmentation mask/image pairs synthesized by FG-DMs trained from scratch (53 M parameters) on the MM-CelebA-HQ, ADE-20K, Cityscapes and COCO datasets.**

(a) MM-CelebA-HQ

(b) ADE-20K

(c) Cityscapes

(d) COCO

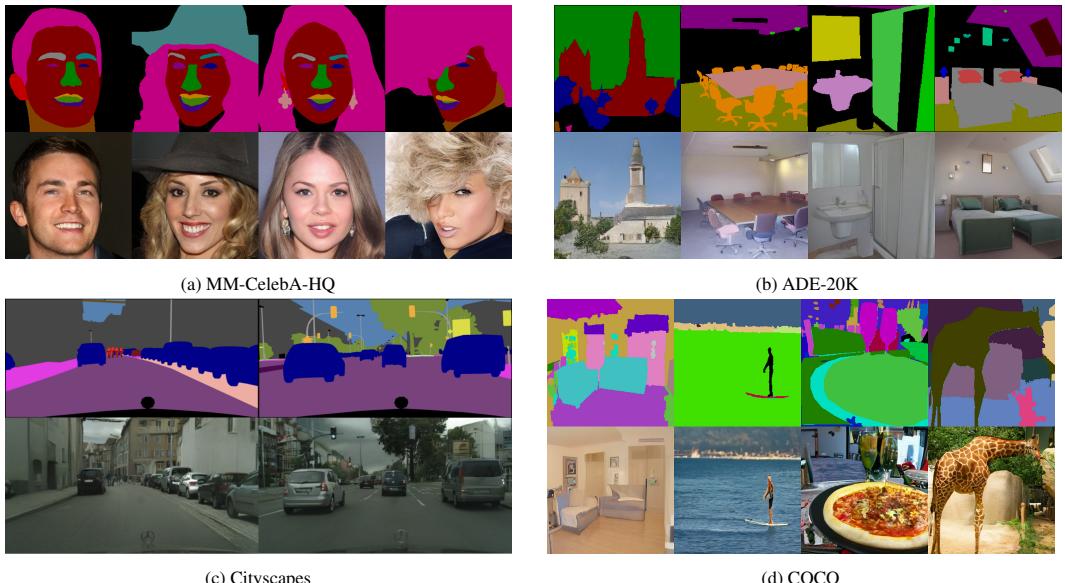



Figure 19: **Semantic Guided conditional image synthesis by FG-DM models trained from scratch (53 M parameters)**. For conditional synthesis, only the image-synthesis VC-DM factor is used. When conditioned by the validation dataset segmentation maps shown at the top, this factor synthesizes the images shown at the bottom.

Fig. 17 shows qualitative results of unconditional synthesis on MM-CelebA-HQ for a model with factors for semantic map, pose and image synthesis. This improves the overall image quality as shown in Table 11. It is observed that the generated pose maps are accurate even for hard examples such as side-view faces.

Figure 18 shows qualitative results of image/segmentation mask pairs generated by FG-DM models trained from scratch on four popular semantic segmentation datasets. The FG-DM can generate good quality samples even on complex datasets such as ADE-20K and COCO using only a small model (53M parameters for each factor).

Figure 19 shows qualitative results of conditional image synthesis with groundtruth maps of validation set on the four datasets using FG-DMs trained from scratch. For conditional synthesis, only the image-synthesis factor of the FG-DM is used. In these experiments, the model is conditioned by the

23

Table 13: **Ablation study on Image Alignment** with the segmentation masks by FG-DM trained separately and jointly.

Model	#P (M)	MM-CelebA		Cityscapes		ADE-20K		COCO	
		mIoU ↑	f.w. IoU ↑	mIoU ↑	f.w. IoU ↑	mIoU ↑	f.w. IoU ↑	mIoU ↑	f.w. IoU ↑
FG-DM (Separate training)	140	70.26	85.77	45.38	81.84	18.07	48.65	23.29	37.33
FG-DM (Joint training)	140	**70.57**	**87.79**	**52.73**	**86.72**	**22.22**	**52.76**	**23.76**	**38.64**

ground-truth validation segmentation masks of the datasets. It is seen that FG-DM can generate high quality samples that align with the latter.

Segmentation-Image Alignment

We quantitatively evaluate the alignment of the generated images with the corresponding segmentation mask using off-the-shelf pretrained models as described in Sec. A.2.5. We compare the image alignment of FG-DM when training separately and jointly. Table 13 shows the results on four datasets where the mIoU score is computed using the groundtruth validation samples for each of the dataset. Once again, FG-DM trained jointly outperforms the FG-DM trained separately indicating the advantage of joint modeling in following the segmentation conditions accurately.

A.2 Implementation details

A.2.1 Extracting COCO Object classes from the prompt using a LLM

We proposed FG-DM for faster sampling of images with high object recall and validated it by by using the groundtruth segmentation maps from the ADE20K validation dataset. For practical use, the object classes can either be manually specified or automatically extracted from the captions which is useful for images involving cluttered scenes with multiple objects. Here, we show that the object classes can be extracted from the caption using a LLM (e.g., chatGPT-3.5).

Specifically, we use the following prompt to elicit responses from chatGPT-3.5.

Prompt

You are ObjectGPT. You will list all possible objects in a scene from the caption description using the set of available classes. The available classes are as follows.
{COCO Classes inserted here as a dictionary mapping the class id to class name.}

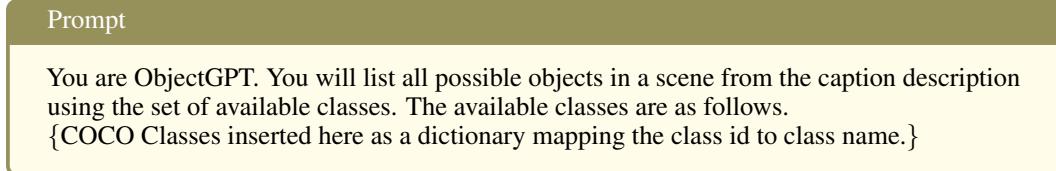



Figure 20a shows example outputs for the two prompts shown in the image. Note that this is zero-shot output where the model is not provided with any example pairs of prompt and corresponding object classes. The accuracy of the task can be further improved by using few-shot in-context examples as shown in the in-context learning (ICL) literature. We show an example of one-shot ICL with one prompt and its corresponding object classes from the COCO validation dataset.

1-shot ICL

"A man is in a kitchen making pizzas"
{Object classes = ["person", "bottle", "cup", "knife", "spoon", "bowl", "broccoli", "carrot", "dining table", "oven", "sink", "branch", "cabinet", "floor-other", "floor-stone", "food-other", "furniture-other", "leaves", "light", "metal", "table", "textile-other", "wall-other", "wall-stone"]}

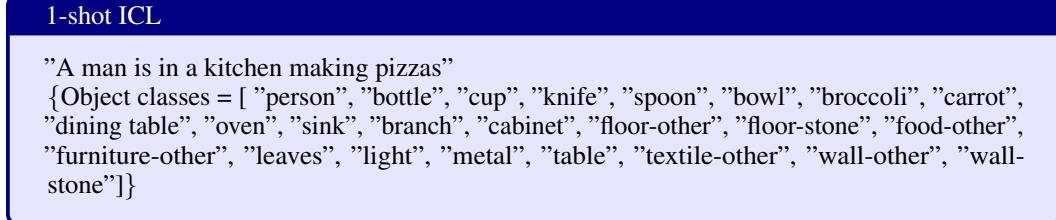
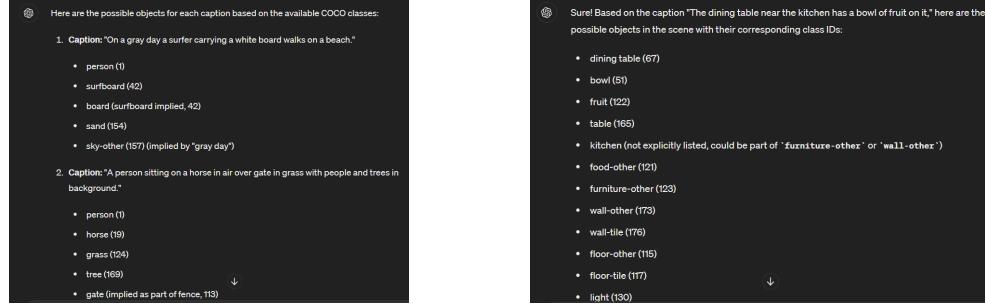



Fig. 20b shows the result for the prompt "The dining table near the kitchen has a bowl of fruit on it." using one-shot ICL shown above.

A.2.2 Segmented Image Editor

We introduce a tool, developed using PyQT to edit the segmentation masks synthesized by the FG-DM, enabling an array of options. As shown in Figure 21, this app has a user-friendly interface for loading and manipulating objects across two distinct segmented maps, which enables very flexible image synthesis. Key features include the ability to add, move, resize or remove objects, flip them, or

24

(a) Extracting COCO object classes from the prompt using chatGPT-3.5 in zero-shot manner.

(b) Extracting COCO object classes from the prompt using 1-shot ICL with chatGPT-3.5

Figure 20: Snapshots of using chatGPT-3.5 to extract the object classes from the input prompt.

replace backgrounds with ease. A unique drawing tool, augmented by a color palette representing 183 objects from the COCO dataset, allows for precise and detailed customization. Furthermore, the app's pointer size adjustment slider for drawing and resizing ensures users can achieve the exact level of size, detail and boldness needed for image editing. Currently, the complexity of these edits is limited by the simplicity of the image editing tool we developed. More complex images will likely be possible with further editing tool development.

(a) Loading the generated segmentation masks into the tool

(b) Resizing the person and moving them to the left of the image

(c) Final edited mask after resizing and moving the airplane

Figure 21: Snapshots of editing capabilities using our segmented image editing tool.

Table 14: Hyperparameter Settings for the FG-DMs trained from scratch on MM-CelebA-HQ, ADE-20K, Cityscapes and COCO datasets. We use an image resolution of 256×512 for Cityscapes and 256×256 for the others.

	MM-CelebA-HQ	ADE-20K	Cityscapes	COCO
f	4	4	4	4
z-shape	$64 \times 64 \times 3$	$64 \times 64 \times 3$	$64 \times 128 \times 3$	$64 \times 64 \times 3$
$|Z|$	8192	8192	8192	8192
Diffusion steps	1000	1000	1000	1000
Optimizer	AdamW	AdamW	AdamW	AdamW
Noise Schedule	linear	linear	linear	linear
Nparams	86M	86M	86M	86M
Channels	128	128	128	128
Depth	2	2	2	2
Channel Multiplier	1,4,8	1,4,8	1,4,8	1,4,8
Attention resolutions	32, 16, 8	32, 16, 8	32, 16, 8	32, 16, 8
Head Channels	32	32	32	32
Batch Size	12	12	12	12
Iterations	632k	632k	93k	632k
Learning Rate	1e-6	1e-6	1.0e-4	1e-6

A.2.3 Hyperparameter Settings

Table 14 summarizes the detailed hyperparameter settings of the FG-DMs trained from scratch reported in the main paper. For FG-DMs adapted from Stable Diffusion, we use the same settings as Stable Diffusion (41) and train only the adapters for 100 epochs with a learning rate of 1e-6 using AdamW optimizer.

25

A.2.4 Training Details

We train all models using 2-4 NVIDIA-A40 GPUs or 2 NVIDIA-A100 GPUs based on the availability. For adapting Stable Diffusion, since we reuse existing conditional model such as ControlNet, we first pretrain the model for 100 epochs to synthesize the conditions (e.g., segmentation, depth, normal or sketch). We then jointly finetune the condition (e.g., segmentation) factor with the conditional image synthesis factor (e.g., ControlNet) for an additional 100 epochs by only updating a subset of parameters of the ControlNet adapter denoted by the `input-hint-block` in the model while keeping the rest frozen. Note that the pretrained ControlNet can still be used in FG-DM without finetuning which results in a slightly lower image quality (also validated in Table 12). For models trained from scratch, we train all the parameters of the U-Net model from scratch.

A.2.5 Evaluation Details

This section provides additional details on evaluation for the experiments of Sec. 4. We follow common practice and estimate the statistics for calculating the FID values (14) shown in Table 7 are based on 10k samples between FG-DM generated samples and the entire training set of each of the datasets. For calculating FID scores we use the torch-fidelity package (31). Following standard practice, we pre-process all the images by resizing to 256×256 for MM-CelebA-HQ, ADE-20K and COCO datasets and 256×512 for Cityscapes dataset for calculating the metrics. Samples are generated with 200 DDIM (45) steps and $\eta = 1.0$. For the measuring the semantic alignment, we use off-the-shelf networks to evaluate the alignment of generated results. We use DRN-D-105 (55) for Cityscapes, ResNet50Dilated-PPM (61) for ADE-20K, Unet (22; 42) for MM-CelebA-HQ and DeepLabV2 (8) for COCO dataset. The generated images are fed to these segmentation models to obtain a pseudo-mask which is compared against the mask which was used to generate the image. We use mean Intersection-overUnion (mIoU) and frequency weighted mIoU to measure the overlap between the generated images and the semantic masks. We calculate the mIoU by upsampling the generated images to the same resolution as default input resolution of the off-the-shelf segmentation models. For computing the metrics, we use the validation segmentation maps for each dataset (3000 images for MM-CelebA-HQ, 2000 images for ADE-20K, 500 images for Cityscapes and 5000 images for COCO) is used. However, it should be noted that this pseudo metric highly depends on the capability of the off-the-shelf network. A strong segmentation network measures the semantic alignment of the generated images more accurately. For all text-to-image synthesis results reported in the paper, we use classifier free guidance with scale 7.5, $\eta = 0.0$ and 20 DDIM steps unless otherwise stated.

A.3 Broader Impact

We introduce a new framework for controlling diffusion models that offers creative image synthesis with higher recall, greater flexibility, modularity and explainability. While it offers the benefits of revealing hidden harmful biases in existing image generative models and offers better interpretability, it can also be potentially misused to propagate harmful, unlawful or unethical information with harmful edits. Since, the framework is modular, any harmful edits can be identified before the image generation step where the segmentation or pose map factor can be filtered (automatically or manually) before proceeding to the image generation factor. Additionally, recent advancements in image watermarking (25) can help to identify generated image contents to protect against these risks.

A.4 Future Work

For future work, FG-DM can be extended for Novel View Synthesis by adding a factor for Novel Views. Further, the Novel View FG-DM with depth/normal factors can be used as a strong prior for controllable Text-to-3D generation with SDS technique (35). The modular nature of FG-DM also allows a potential extension for audio/video generation making FG-DM framework to be a strong candidate for multi-modal content generation.

26

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The major claims made by the paper regarding the object recall, image editing capabilities and data augmentation are justified by experiments.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The conclusion section discusses the limitations of the current approach and possible future work.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

27

Answer: [NA]

Justification: There are no theorems or proofs introduced in the paper.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The hyperparameter settings, GPU configurations and auxiliary models employed by the work are detailed in the appendix which is sufficient to reproduce the results presented in the paper.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

28

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Code and trained models will be released on the project page. FG-DM

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Appendix lists down the detailed hyperparameter configurations for all models trained for this paper.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Following the literature, evaluations are reported consistent with prior works where the error bars are not shown.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

29

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Detailed compute resources and the GPUs used for training islisted in the Appendix

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

Answer: [Yes]

Justification: The paper followsthe code of ethics listed in the guidelines.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Limitations and Broader impacts in Appendix discuss this

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

30

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: Image generators will be released with the safety filters

    Guidelines:
    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All prior works and prior models used in this work are cited in the paper.

    Guidelines:
    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

31

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: No new assets introduced.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: No human subject is involved.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: Does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

32