Unlocking Pre-trained Image Backbones for Semantic Image Synthesis

Tariq Berrada[1,2] Jakob Verbeek[1] Camille Couprie[1] Karteek Alahari[2]
[1]FAIR, Meta [2]Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK

Figure 1. Images generated with models trained on COCO-Stuff. We compare our approach to state-of-the-art methods OASIS, SDM, and PITI, along with inference times to generate a single image. Our approach combines high-quality samples with low-latency sampling.

Abstract

Semantic image synthesis, i.e., generating images from user-provided semantic label maps, is an important conditional image generation task as it allows to control both the content as well as the spatial layout of generated images. Although diffusion models have pushed the state of the art in generative image modeling, the iterative nature of their inference process makes them computationally demanding. Other approaches such as GANs are more efficient as they only need a single feed-forward pass for generation, but the image quality tends to suffer when modeling large and diverse datasets. In this work, we propose a new class of GAN discriminators for semantic image synthesis that generates highly realistic images by exploiting feature backbones pre-trained for tasks such as image classification. We also introduce a new generator architecture with better context modeling and using cross-attention to inject noise into latent variables, leading to more diverse generated images. Our model, which we dub DP-SIMS, achieves state-of-the-art results in terms of image quality and consistency with the input label maps on ADE-20K, COCO-Stuff, and Cityscapes, surpassing recent diffusion models while requiring two orders of magnitude less compute for inference.

1. Introduction

Conditional image synthesis aims to generate images based on information such as text, categories, sketches, label maps, *etc*. While text-based generation has seen impressive advances in recent years with diffusion models [41, 49], it lacks precise control over the location and the boundaries of objects, which are important properties for creative content generation tasks like photo editing, inpainting, and for data augmentation in discriminative learning [1, 4, 18, 68]. Consequently, in this work we focus on semantic image synthesis [24, 44, 53, 59–61], where the goal is to produce an image, given a segmentation map, with every pixel assigned to a category, as input. Due to the one-to-many nature of the mapping, prior works have tackled this problem in a conditional GAN [17] framework by exploring different conditioning mechanisms in GANs to do stochastic generations that correspond to the input label map [24, 44, 60]. Others developed conditional discriminator models, which avoid image-to-image reconstruction losses that compromise diversity in generated images [53]. Diffusion models [59, 61] have also been investigated for this problem. SDM [61] adds spatially adaptive normalization layers for conditioning, while PITI [59] replaces the text encoder of a pre-trained text-to-image diffusion model. In comparison to GANs, diffusion models often result in improved image quality, but suffer from lower consistency with the input segmentation maps, and are slower during inference due to the iterative sampling process [7].

To improve the image quality and consistency of GAN-based approaches, we explore the use of pre-trained image backbones in discriminators for semantic image synthesis. Although leveraging pre-trained image models is common in many other vision tasks, such as classification, segmentation, or detection, and more recently for class-conditional GANs [52], to our knowledge this has not been explored for semantic image synthesis. To this end, we develop a UNet-like encoder-decoder architecture where the encoder is a fixed pre-trained image backbone, which leverages the multi-scale feature representations embedded therein, and the decoder is a convolutional residual network. We also propose a novel generator architecture, building on the dual-pyramid modulation approach [34] with an improved label map encoding through attention mechanisms for better diversity and global coherence among the images generated. Finally, we add contrastive and diversity losses to further improve the quality and diversity of generated images.

We validate our contributions with experiments on the ADE-20K, COCO-Stuff, and Cityscapes datasets. Our model, termed *DP-SIMS* for "Discriminator Pre-training for Semantic IMage Synthesis", achieves state-of-the-art performance in terms of image quality (measured by FID) and consistency with the input segmentation masks (measured by mIoU) across all three datasets. Our results not only surpass recent diffusion models on both metrics, but also come with two orders of magnitude faster inference.

In summary, our main contributions are the following:

- We develop an encoder-decoder discriminator that leverages feature representations from pre-trained networks.
- We propose a generator architecture using attention mechanisms for noise injection and context modeling.
- We outperform state-of-the-art GAN and diffusion-based methods in image quality, input consistency, and speed.

2. Related work

Generative image modeling. Several frameworks have been explored in deep generative modeling, including GANs [17, 22, 24, 25, 28, 44, 53], VAEs [30, 47, 58], flow-based models [14, 15, 29] and diffusion-based models [13, 21, 49, 59, 61]. GANs consist of generator and discriminator networks which partake in a mini-max game that results in the generator learning to model the target data distribution. GANs realized a leap in sample quality, due to the mode-seeking rather than mode-covering nature of their objective function [38, 42]. More recently, breakthrough results in image quality have been obtained using text-conditioned diffusion models trained on large-scale text-image datasets [2, 16, 41, 46, 49]. The relatively low sampling speed of diffusion models has triggered research on scaling GANs to training on large-scale datasets to achieve competitive image quality while being orders of magnitude faster to sample [25].

Semantic image synthesis. Early approaches for semantic image synthesis leveraged cycle-consistency between generated images and conditioning masks [24, 60] and spatially adaptive normalization (SPADE) layers [44]. These approaches combined adversarial losses with image-to-image feature-space reconstruction losses to enforce image quality as well as consistency with the input mask [67]. OASIS [53] uses a UNet discriminator model which labels pixels in real and generated images with semantic classes and an additional "fake" class, which overcomes the need for feature-space losses that inherently limit sample diversity, while also improving consistency with the input segmentation maps. Further improvements have been made by adopting losses to learn image details at varying scales [34], by exploiting intermediate representations such as edges to guide the generation process [57], or through multi-modal approaches which leverage data from different modalities like text, sketches and segmentations [22].

Several works have explored diffusion models for semantic image synthesis. SPADE layers were incorporated in the denoising network of a diffusion model in SDM [61] to align the generated images with semantic input maps. PITI [59] replaced the text-encoder of pre-trained text-to-image diffusion models, with a label map encoder, and fine-tuned the resulting model. FLIS [63] propose a rectified cross-attention module which integrates unseen semantic masks into the diffusion process of large-scale text-to-image pre-trained diffusion models. In our work, rather than relying on generative pre-training as in PITI and FLIS, we leverage discriminative pre-training.

Another line of work considers generating images from segmentation maps with free-text annotations [3, 12, 63]. These diffusion approaches, however, exhibit relatively poor consistency with the input label maps while also being slower to sample from than GAN-based models.

Pre-trained backbones in GANs. Pre-trained feature representations have been explored in various ways in GAN training. When the model is conditioned on detailed inputs, such as sketches or segmentation maps, pre-trained backbones are used to define a reconstruction loss between the generated and training images [67]. Another line of work leverages these backbones as fixed encoders in adversarial discriminators [48, 52]. Naively using a pre-trained encoder with a fixed decoder yields suboptimal results, thus the projected GANs model [52] uses a feature conditioning strategy based on random projections to make the adversarial game more balanced. While this approach is successful

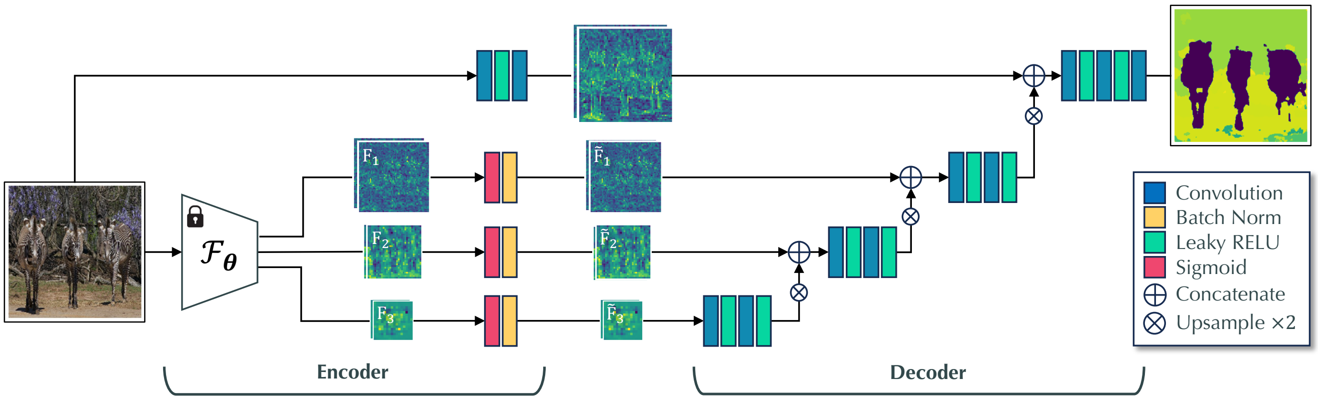

Figure 2. **Architecture of our discriminator model.** The encoder consists of a pre-trained feature backbone $\mathcal{F}_{\theta}$ (left), residual blocks at full-image resolution (top), and trained feature decoder that aggregates the multi-scale features from the frozen backbone (right).

with some backbones, the method worked best with small pre-trained models such as EfficientNets [56], while larger models resulted in lower performance. A related line of work [32] uses an ensemble of multiple pre-trained backbones to obtain a set of discriminators from which a subset is selected at every step for computing the most informative gradients. This produced impressive results but has the following significant overheads which make it inefficient: (i) all the discriminators and their associated optimizers are stored in memory, (ii) there is a pre-inference step to quantify the suitability of each discriminator for any given batch, and (iii) the main discriminator is trained from scratch. Our work is closely related to projected GANs, but to our knowledge the first one to leverage pre-trained discriminative feature networks for semantic image synthesis.

Attention in GANs. While most of the popular GAN frameworks, such as the StyleGAN family, relied exclusively on convolutions [26–28], some other works explored the use of attention in GANs to introduce a non-local parametrization that operates beyond the receptive field of the convolutions in the form of self-attention [5, 23, 25, 33, 66], as well as cross-attention to incorporate information from different modalities (text-to-image). To the best of our knowledge, our work is the first to explore cross-attention layers in semantic image synthesis models.

3. Method

Semantic image synthesis aims to produce realistic RGB images $\mathbf{g} \in \mathbb{R}^{W \times H \times 3}$ that are consistent with an input label map $\mathbf{t} \in \mathbb{R}^{W \times H \times C}$, where C is the number of semantic classes and $W \times H$ is the spatial resolution. A one-to-many mapping is ensured by conditioning on a random noise vector $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ of dimension d_z.

In this section, we present our GAN-based approach, starting with our method to leverage pre-trained feature backbones in the discriminator (Sec. 3.1). We then describe our noise injection and label map modulation mechanisms for the generator (Sec. 3.2), and detail the losses we use to train our models (Sec. 3.3).

3.1. Pre-trained discriminator backbones

Our discriminator is an *encoder-decoder* model where the decoder is made of residual blocks with skip connections similar to [50, 53], while the encoder is a fixed and pre-trained feature backbone network followed by a feature conditioning module. The discriminator is trained to classify pixels as belonging to their semantic category or an additional "fake" class for synthetic images.

Let $\mathcal{F}_{\theta}$ be a pre-trained feature backbone with parameters θ. We use this backbone, frozen, as part of the "encoder" in the UNet discriminator. Let $\mathbf{F}_l \in \mathbb{R}^{C_l \times W_l \times H_l}$ denote the features extracted by the backbone at levels $l = 1, \ldots, L$, which generally have different spatial resolutions $W_l \times H_l$ and number of channels C_l. These features are then processed by the UNet "decoder", which is used to predict per-pixel labels spanning the semantic categories present in the input label map, as well as the "fake" label. Additionally, to exploit high-frequency details in the image, we add a fully trainable path at the full-image resolution with two relatively shallow residual blocks. The full discriminator architecture is illustrated in Fig. 2.

Feature conditioning. An important problem with using pre-trained backbones is feature conditioning. Typical backbones are ill-conditioned, meaning that some features are much more prominent than others. This makes it difficult to fully exploit the learned feature representation of the backbone as strong features overwhelm the discriminator's decoder and result in exploring only certain regions in the feature representation of the encoder. Previously, [52] tried to alleviate this problem by applying cross-channel mixing (CCM) and cross-scale mixing (CSM) to the features, while [32] average the signals from multiple discriminators

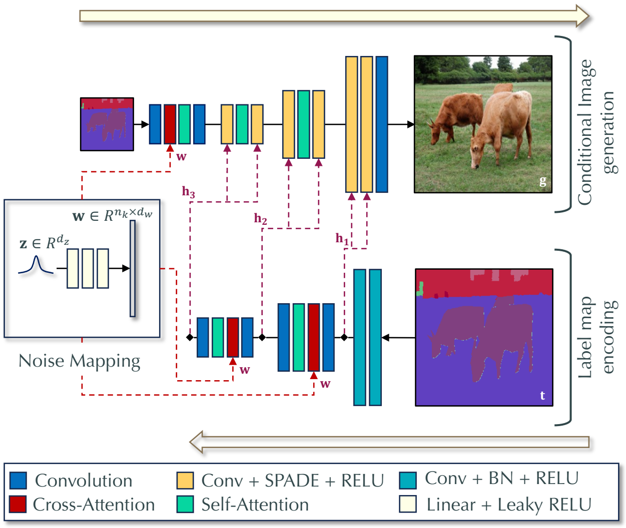

Figure 3. **Our generator architecture consist of two components.** (i) A conditional image generation network (top) that takes a low-resolution label map as input and produces the full-resolution output image. (ii) A semantic map encoding network (bottom) that takes the full resolution label map as input and produces multi-scale features that are used to modulate the intermediate features of the image generation network.

to obtain a more diluted signal. Empirically, the first approach underperforms in many of our experiments, as the strong features still tend to mask out their weaker, yet potentially relevant, counterparts. On the other hand, the second introduces a large overhead from the multiple models being incorporated in training. In our work, we develop a method that better exploits the feature representation from the encoder. We achieve this by aiming to make all features have a comparable contribution to the downstream task.

Consider a feature map $\mathbf{F}_l \in \mathbb{R}^{C_l \times W_l \times H_l}$ at scale l from the pre-trained backbone. First, we apply a contractive non-linearity (CNL) such as sigmoid to obtain $\mathbf{F}_l' = \sigma(\mathbf{F}_l)$. Next, we normalize the features to ensure they have a similar contribution in the following layers. We choose batch normalization, yielding $\tilde{\mathbf{F}}_l = (\mathbf{F}_l' - \mu_l)/\sigma_l$, where μ_l and σ_l are the batch statistics. In this manner, all features are in a similar range and therefore the decoder does not prioritize features with a high variance or amplitude.

3.2. Generator architecture

Our generator architecture is based on DP-GAN [34], but offers two main novelties: a revisited noise injection mechanism and improved modeling of long-range dependencies through self-attention. Following DP-GAN, we use a mask encoding network to condition the SPADE blocks, rather than conditioning the SPADE blocks on the label maps via a single convolution layer, which cannot take into account longer-range dependencies encoded in the label map. Each block of the label map encoding pyramid is made of a single convolution layer with downsampling followed by batch norm, GELU activation [19], attention modules, and a pointwise convolution layer. For every scale, we obtain a modulation map $\mathbf{h}_i, i \in \{1, \ldots, L\}$ which, concatenated with a resized version of the ultimate map $\mathbf{h}_L$, will serve as conditioning for the SPADE block at the same resolution.

While [53] argued that concatenating a spatial noise map to the label map was enough to induce variety in the generated images, since the noise is present in all SPADE blocks, and therefore hard to ignore, the same cannot be said for the architecture of DP-GAN [34]. The noise is injected only at the first layer of the label map encoding network, hence it is much easier to ignore. Consequently, we propose a different mechanism for noise injection, making use of cross-attention between the learned representations at different scales and the mapping noise obtained by feeding $\mathbf{z}$ to a three-layer MLP, $\mathbf{w} = \text{MLP}(\mathbf{z}) \in \mathbb{R}^{n_k \times d_w}$. Let $\mathbf{h}_i \in \mathbb{R}^{C_i \times H_i \times W_i}$ be the downsampled feature representation from the previous scale. This feature $\mathbf{h_i}$ first goes through a convolution to provide an embedding of the label map, then the spatial dimensions are flattened and projected via a linear layer to obtain the queries $Q \in \mathbb{R}^{H_i W_i \times d_q}$. The transformed noise vector $\mathbf{w}$ is projected via two linear layers to obtain the keys and the values $K, V \in \mathbb{R}^{n_k \times d_q}$, then the cross-attention is computed as:

$$\mathbf{A} = \text{SoftMax}\left(QK^\top / \sqrt{d_q}\right) V. \tag{1}$$

The noise injection blocks at spatial resolutions 64×64 and lower use residual cross-attention block

$$a(\mathbf{h}_i, \mathbf{w}) = \mathbf{h}_i + \eta_i \cdot \mathbf{A}(\mathbf{h}_i, \mathbf{w}), \tag{2}$$

where $\eta_i \in \mathbb{R}$ is a trainable gating parameter initialized at 0. Noise injection is followed by a residual self-attention block, before having a convolution output the conditioning at scale i. For higher resolutions where attention modules are too expensive, we use convolutional blocks only. The generator architecture is illustrated in Fig. 3.

3.3. Training

We train our models by minimizing a weighted average of three loss functions which we detail below.

Pixel-wise focal loss. Our main loss is based on a pixel-wise GAN loss [53], where the discriminator aims to assign pixels in real images to the corresponding class in the conditioning label map, and those in generated images to an additional "fake" class. To improve the performance on rare classes, we replace the weighted cross-entropy of [53] with a weighted focal loss [35], while keeping the same weighting scheme as in [53]. Let $p(\mathbf{x}) \in [0, 1]^{H \times W \times (C+1)}$ denote the output class probability map of the discriminator for a real RGB image $\mathbf{x}$, and $p(\mathbf{g}) \in [0, 1]^{H \times W \times (C+1)}$ be the probability map for a generated image $\mathbf{g} = G(\mathbf{z}, \mathbf{t})$, where the label index $C+1$ is used for the “fake” class. Then, the discriminator loss is:

$$\begin{aligned}\mathcal{L}_{\mathrm{D}} = -\mathbb{E}_{(\mathbf{x},\mathbf{t})} \sum_{c=1}^{C} \alpha_c \sum_{i=1}^{H\times W} \mathbf{t}_{i,c} \left(1 - p(\mathbf{x})_{i,c}\right)^{\gamma} \log p(\mathbf{x})_{i,c} \\ -\mathbb{E}_{(\mathbf{g},\mathbf{t})} \sum_{i=1}^{H\times W} \left(1 - p(\mathbf{g})_{i,C+1}\right)^{\gamma} \log p(\mathbf{g})_{i,C+1}, \quad (3)\end{aligned}$$

where α_c's are the class weighting terms and γ is a hyper-parameter of the focal loss. The standard cross-entropy is recovered for $\gamma = 0$, and for $\gamma > 0$ the loss puts more weight on poorly predicted labels.
The pixel-wise loss for the generator then takes the form:

$$\mathcal{L}_{\mathrm{G}} = -\mathbb{E}_{(\mathbf{g},\mathbf{t})} \sum_{c=1}^{C} \alpha_c \sum_{i=1}^{H\times W} \mathbf{t}_{i,c} \left(1 - p(\mathbf{g})_{i,c}\right)^{\gamma} \log p(\mathbf{g})_{i,c}. \quad (4)$$

Using the focal loss, both the generator and discriminator put more emphasis on pixels that are incorrectly classified. These often belong to rare classes which helps to improve performance for these under-represented classes. To prevent the discriminator output probabilities from saturating and thus leading to vanishing gradients, we apply one-sided label smoothing [51] by setting the cross-entropy targets to $1-\epsilon$ for the discriminator loss, where ϵ is a hyper-parameter.

Contrastive loss. We define a patch-wise contrastive loss that encourages the generated images to be globally coherent. Our contrastive framework is based on InfoNCE [43], which aims to bring matching patch features closer together, and push them further from non-matching features. Given a pair $(\mathbf{x}, \mathbf{t})$ of image and label map, we generate a corresponding image $\mathbf{g} = G(\mathbf{z}, \mathbf{t})$, and use $\mathbf{H}_{\mathbf{x}}$ and $\mathbf{H}_{\mathbf{g}}$ the corresponding multi-scale features obtained from a pre-trained VGG network [54]. For every scale, we sample matching features $\mathbf{z}, \mathbf{z}^+$ from the same spatial coordinates in $\mathbf{H}_{\mathbf{g}}$ and $\mathbf{H}_{\mathbf{x}}$ respectively. Additionally, we sample N non-matching features $\mathbf{z}_n^-$ at randomly selected coordinates from $\mathbf{H}_{\mathbf{x}}$.

The features are then projected into an embedding space using a convolution followed by a two-layer MLP to obtain $\mathbf{v}, \mathbf{v}^+, \mathbf{v}_n^- \in \mathbb{R}^{d_v}$ before computing the InfoNCE loss as

$$\mathcal{L}_{\mathrm{NCE}}(\mathbf{v}, \mathbf{v}^+, \mathbf{v}^-) = -\log\left(\frac{e^{\mathbf{v}^\top \mathbf{v}^+/\tau}}{e^{\mathbf{v}^\top \mathbf{v}^+/\tau} + \sum_{n=1}^{N} e^{\mathbf{v}^\top \mathbf{v}_n^-/\tau}}\right), \quad (5)$$

where τ is a temperature parameter controlling the sharpness in the response of the loss. We apply the loss at feature scales $1/4, 1/8, 1/16$, and take their sum. This is similar to the contrastive losses used for image-to-image translation [45], with the main difference being the feature representation from which the loss is calculated. While other methods reuse the encoder features from their translation network, we obtain the feature pyramid from a VGG network [54] and process it by a simple module made of a convolution block followed by a projection MLP.

Diversity loss. To promote diversity among the generated images we introduce a loss, similar to [39, 64], that encourages two images generated with the same mask, but different latents $\mathbf{z}$, to be sufficiently distinct from each other:

$$\mathcal{L}_{\mathrm{Div}} = \max\left[0, \tau_{\mathrm{div}} - \frac{\left\|G^f(\mathbf{z}_1, \mathbf{t}) - G^f(\mathbf{z}_2, \mathbf{t})\right\|_1}{\|\mathbf{z}_1 - \mathbf{z}_2\|_1}\right], \quad (6)$$

where G^f is the feature output of the generator before the final convolution. We adopt a cutoff threshold τ_{div} on the loss in order to not overly constrain the generator, and apply this loss only for similar samples given the same label map.

4. Experiments

We present our experimental setup in Sec. 4.1, followed by our main results in Sec. 4.2, and ablations in Sec. 4.3.

4.1. Experimental setup

Datasets. We consider three popular datasets to benchmark semantic image synthesis: COCO-Stuff [6], Cityscapes [11], ADE-20K [69]. COCO-Stuff provides 118k training images and 5k validation images, labeled with 183 classes. Cityscapes contains 2,975 training images along with a validation set of 500 images, and uses 35 labels. ADE-20K holds 26k images with object segmentations across 151 classes. Similar to [44, 60, 61], we use instance-level annotations when available. For COCO-Stuff and Cityscapes, we use instance segmentations as in [9], by creating vertical and horizontal offset maps of every foreground pixel w.r.t. its object center of mass, and concatenate these to the semantic label maps as input for the model. For ADE-20K, there are no instance segmentations available. We generate images at a resolution of 256×256 for ADE-20K and COCO-Stuff, and 256×512 for Cityscapes. We blurred faces of people in the datasets before use; see the supplementary material for more details.

Metrics. We compute FID [20] to assess image quality, and report the mean intersection-over-union score (mIoU) to measure the consistency with the input segmentation maps. For a fair comparison with previous work [34, 44, 53], we used the segmentation models from these works for inferring label maps of the generated images: UperNet101 [62] for ADE-20K, multi-scale DRN-D-105 [65] for Cityscapes, and DeepLabV2 [8] for COCO-Stuff. We refer to the scores obtained with these models as mIoU. In addition, we infer label masks using Mask2Former [10], which is more accurate than other segmentation models, thus yielding a more meaningful comparison to the ground-truth masks. We denote the resulting scores as $\mathrm{mIoU}_{\mathrm{MF}}$. See the supplementary material for more detail.

	COCO			ADE20k			Cityscapes		
	FID (↓)	mIoU_{MF} (↑)	mIoU (↑)	FID (↓)	mIoU_{MF} (↑)	mIoU (↑)	FID (↓)	mIoU_{MF} (↑)	mIoU (↑)
Pix2pixHD [60]	111.5	—	14.6	73.3	—	22.4	104.7	—	52.4
SPADE [44]	22.6	—	37.4	33.9	—	38.5	71.8	—	62.3
OASIS [53]	17.0	52.1	44.1	28.3	53.5	48.8	47.7	72.0	69.3
DP-GAN [34]	—	—	—	26.1	—	52.7	44.1	—	73.6
PoE-GAN [22]	15.8	—	—	—	—	—	—	—	—
ECGAN++ [57]	14.9	—	47.9	24.7	—	52.7	42.2	—	73.3
SDM [61]	15.9	40.3	36.8	27.5	51.9	44.0	42.1	72.8	69.1
PITI [59]	15.5	31.2	29.5	—	—	—	—	—	—
FLIS [63]	14.4	—	40.7	25.0	—	41.9	—	—	—
DP-SIMS (ours)	**13.6**	**65.2**	**57.7**	**22.7**	**67.8**	**54.3**	**38.2**	**78.5**	**76.3**

Table 1. Comparison of DP-SIMS to state-of-the-art GAN-based (first block) and diffusion-based methods (second block). Results taken from the original papers. We computed the mIoU_{MF} metric for methods where pre-trained checkpoints or generated images are available.

Backbone	Prms.	FLOPS	Acc@1	FID (↓)	mIoU_{MF} (↑)
Swin-B	107 M	15.4G	**86.4**	29.5	55.4
ResNet-50	44 M	4.1G	76.2	24.6	60.5
EfficientNet-Lite1	3 M	631M	83.4	24.5	63.1
ConvNeXt-B [37]	89 M	15.4G	85.1	23.5	63.5
ConvNeXt-L [37]	198 M	34.4G	85.5	**22.7**	**67.8**

Table 2. Comparison of backbone architectures on ADE-20K. We report the ImageNet-1k top-1 accuracy (Acc@1) for reference.

Pre-training	Acc@1	FID (↓)	mIoU_{MF} (↑)
Random Init.	—	52.9	40.7
IN-1k@224	84.3	**22.7**	62.8
IN-21k@224	86.6	23.6	64.1
IN-21k@384	87.5	**22.7**	**67.8**

Table 3. Influence of discriminator pre-training on the overall performance for ADE-20K using a ConvNext-L backbone.

Implementation details. We counter the strong class imbalance in the datasets used in our experiments with a sampling scheme favoring rare classes. Let f_c be the fraction of training images where class c appears, then each image is sampled with a probability proportional to $f_k^{-1/2}$ with k the sparsest class present in the image.

Each of our models is trained on one or two machines with eight V100 GPUs. We set the total batch size at 64 and use ADAM optimizer in all our experiments with a learning rate of 10^{-3} and momentums $\beta_1 = 0, \beta_2 = 0.99$. For pre-trained Swin backbones, we found it necessary to use gradient clipping to stabilize training. Following prior work, we track an exponential moving average of the generator weight and set the decay rate to $\alpha = 0.9999$. For the contrastive loss, we set the weighting factor $\lambda_C = 100$, the temperature $\tau = 0.3$ and select $N = 128$ negative samples. We set $\lambda_{\text{GAN}} = 1$ for the GAN loss and $\lambda_D = 10$ for the diversity loss. For the focal loss, we set $\gamma = 2$.

4.2. Main results

Comparison to the state of the art. In Tab. 1, we report the results obtained with our model in comparison to the state of the art. Our DP-SIMS method (with ConvNext-L backbone) achieves the best performance across metrics and datasets. On COCO-Stuff, we improve the FID of 14.4 from FLIS [63] to 13.6, while improving the mIoU_{MF} of 52.1 from OASIS to 65.2, and mIoU of 47.9 from ECGAN++ to 57.7. For ADE-20K, we observe a similar trend with an improvement of 2.0 FID points w.r.t. ECGAN++, an improvement of 14.5 points in mIoU_{MF} w.r.t. OASIS, and improving mIoU by 1.6 points w.r.t. ECGAN++ and DP-GAN. For Cityscapes, we obtain improvements of 3.9 FID points w.r.t. Semantic Diffusion, 5.5 points in mIoU_{MF}, and 3.0 points in mIoU. See Fig. 1 and Fig. 4 for qualitative comparisons of model trained on COCO-Stuff and Cityscapes. Please refer to the supplementary material for additional examples, including ones for ADE-20K.

Encoder architecture. We experiment with different pre-trained backbone architectures for the discriminator in Tab. 2. All the encoders were trained for ImageNet-1k classification. We find that the attention-based Swin architecture [36] has the best ImageNet accuracy, but compared to convolutional models performs worse as a discriminator backbone for semantic image synthesis, and tends to be more unstable, often requiring gradient clipping to converge. For the convolutional models, better classification accuracy translates to better FID and mIoU_{MF}.

Pre-training dataset. In Tab. 3, we analyze the impact of pre-training the ConvNext-L architecture in different ways and training our models on top of these, with everything else being equal. We consider pre-training on ImageNet-1k (IN-1k@224) and ImageNet-21k (IN-21k@224) at 224×224 resolution, and also on ImageNet-21k at 384×384 resolution (IN-21k@384). In terms of mIoU_{MF}, the results are in line with those observed for different architectures: discriminators trained with larger datasets (IN-21k) and on higher resolutions perform the best. On the other hand, we find that for FID, using the standard ImageNet (IN-1k@224) results in better performance than its bigger IN-21k@224 coun-

Figure 4. Qualitative comparison with prior work on the Cityscapes dataset. We show the results of OASIS [53], SDM [61], and our approach along with the corresponding label map used for generating each image. Note that our method generates more coherent objects with realistic textures in comparison.

Model	Gen. steps	Ups. steps	Δt_{gen}	Δt_{ups}	Δt_{tot}
PITI	250	27	14.3	3.1	17.4
PITI	27	27	1.5	3.1	4.6
SDM	1000	—	260.0	—	260.0
DP-SIMS	—	—	—	—	0.04

Table 4. Comparison of inference time (in seconds) of PITI, SDM and our GAN-based model. We show the time taken by the generative (Δt_{gen}) and the upsampling (Δt_{ups}) models in addition to the total time (Δt_{tot}) for these steps.

	EfficientNet-Lite1		ConvNeXt-L	
	FID	mIoU_{MF}	FID	mIoU_{MF}
Baseline - no normalization	27.8	58.6	24.4	63.6
CCM + CSM (PG)	28.9	59.1	25.4	66.0
BatchNorm + CCM + CSM	29.4	56.4	24.6	66.4
DP-SIMS w/o sigmoid	24.9	62.7	23.3	65.7
DP-SIMS w/o BatchNorm	26.0	61.6	23.6	64.0
DP-SIMS (ours)	**24.5**	**63.1**	**22.7**	**67.8**

Table 5. Ablation on feature conditioning shown on ADE-20K with two backbones.

terpart, and performs as well as IN-21k@384 pre-training. This is likely due to the use of the same dataset in the Inception model [55], which is the base for calculating FID, thus introducing a bias in the metric.

Inference speed. An important advantage of GAN models over their diffusion counterparts is their fast inference. While a GAN only needs one forward pass to generate an image, a diffusion model requires several iterative denoising steps, resulting in slower inference, which can hamper the practical usability of the model. In Tab. 4 we report the inference speed for generating a single 256×256 image, averaged over 50 different runs. PITI uses 250 denoising steps for the generative model at 64×64 resolution and 27 steps for the upsampling model by default, while SDM uses 1000 steps at full resolution. We also benchmark using 27 steps for the PITI generative model. Our generator is two to three orders of magnitude faster than its diffusion counterparts.

4.3. Ablations

Feature Conditioning. We perform an ablation to validate our feature conditioning mechanisms on ADE-20K in Tab. 5. We compare to: (i) a baseline that does not normalize the backbone features, (ii) the cross-channel and scale mixing approach of Projected GAN (PG) [52], (iii) using our BatchNorm layer with cross-channel and scale mixing, (iv) DP-SIMS without the sigmoid normlization, (v) DP-SIMS without the BatchNorm layers. For a fair comparison with [52], these experiments are conducted on their best reported backbone, EfficientNet-Lite1 [56]. We also conducted this experiment with a ConvNeXt-L backbone. Compared to the baseline, Projected GAN improves mIoU_{MF} but degrades FID, while our feature conditioning (last line) improves both metrics for both backbones. Moreover, the ablations show that both the sigmoid and BatchNorm contribute, and that adding BatchNorm for ProjectedGAN leads to inferior performance.

	FID (↓)	mIoU_{MF} (↑)
DP-SIMS	**22.7**	**67.8**
Generator architecture		
OASIS disc + our gen	29.3	49.0
OASIS gen + our disc	25.6	63.6
Ours w/o self-attention	23.7	65.4
Ours w/o cross-attention	23.6	64.5
Training		
Ours w/o label smoothing	23.0	66.3
Ours w/o contrastive loss	25.1	66.0

Table 6. Ablations on the architectural design and training losses, shown on ADE-20K with ConvNext-L backbone.

τ	0.07	0.3	0.7	2.0
FID	25.7	**22.7**	24.1	26.4
mIoU_{MF}	62.6	**67.8**	66.3	61.4

Table 7. Influence of the contrastive loss evaluated on ADE-20K.

	Cityscapes		ADE-20K	
	FID (↓)	mIoU_{MF}(↑)	FID (↓)	mIoU_{MF}(↑)
Weighted CE	39.8	75.9	23.2	65.5
Focal	39.3	75.0	22.8	64.7
Weighted focal	**38.2**	**78.5**	**22.7**	**67.8**

Table 8. Comparison of pixel-wise losses on Cityscapes and ADE-20K with ConvNext-L backbone.

Architectural modifications. In Tab. 6, we perform an ablation on our proposed architectural modifications. Swapping out our generator or discriminator with the ones from OASIS, suggests that most of the gains are due to our discriminator design. Using the OASIS discriminator instead of ours deteriorates mIoU_{MF} by 18.8 points and FID by 6.6 points. We also experiment with removing the cross-attention noise injection mechanism and replacing it with the usual concatenation instead, as well as leaving out the self-attention layers. Both of these contribute to the final performance in a notable manner. Finally, we present an ablation on label smoothing, which deteriorates FID by 0.3 and $\text{mIoU}_{\text{MF}}$by 1.4 points when left out.

Contrastive loss. To assess the importance of the contrastive loss, we perform an ablation in the last row of Tab. 6 where we remove it during training. This substantially impacts the results: worsening FID by 2.4 and mIoU_{MF} by 1.8 points. In Tab. 7, we evaluate different values for the temperature parameter τ, and find an optimal temperature parameter $\tau_C = 0.3$, using $\lambda_c = 100$.

Focal loss. In Tab. 8, we consider the impact of the focal loss by comparing it to the weighted cross-entropy loss, as used in OASIS, and the effect of class weighting in the focal loss. We find that for both datasets switching from weighted cross-entropy to the focal loss improves FID but worsens mIoU_{MF}. The weighted focal loss, however, improves both metrics on both datasets.

Diversity. We study the effect of the diversity loss on the variability of generated images. Following [53], we report the mean LPIPS distance across 20 synthetic images from the same label map, averaged across the validation set, in Tab. 9. A qualitative example is provided in Fig. 5 showing a clear variety in the images generated. In comparison with OASIS, we generate more diverse images, with an LPIPS score similar to that of SPADE, but with a much higher quality, as reported in Tab. 1, in terms of FID and mIoU_{MF}.

Model	3D noise	LPIPS (↑)
SPADE+	✓	0.16
SPADE+	✗	0.50
OASIS	✓	0.35
DP-SIMS	✗	0.47

Table 9. Evaluation of the diversity of images generated. Results on ADE-20K for SPADE+ and OASIS are taken from [53].

Figure 5. Images generated by varying the noise vector with DP-SIMS trained on COCO-Stuff and using a ConvNext-L backbone.

5. Conclusion

We introduced DP-SIMS that harnesses pre-trained backbones in GAN-based semantic image synthesis models. We achieve this by using them as an encoder in UNet-type discriminators, and introduce a feature conditioning approach to maximize the effectiveness of pre-trained features. Moreover, we propose a novel generator architecture which uses cross-attention to inject noise in the image generation process, and introduce new loss terms to boost sample diversity and input consistency. We experimentally validate our approach and compare it to state-of-the-art prior work based on GANs as well as diffusion models on three standard benchmark datasets. Compared to these, we find improved performance in terms of image quality, sample diversity, and consistency with the input segmentation maps. Importantly, with our approach inference is two orders of magnitude faster than diffusion-based methods.

In our experiments we found that transformer-based models, such as Swin, can lead to instability when used as discriminator backbones. Given their strong performance for dense prediction tasks, it would be worthwhile to further study and mitigate this issue in future work, hopefully bringing additional improvements.

References

[1] Mohamed Akrout, Bálint Gyepesi, Péter Holló, Adrienn Poór, Blága Kincső, Stephen Solis, Katrina Cirone, Jeremy Kawahara, Dekker Slade, Latif Abid, Máté Kovács, and István Fazekas. Diffusion-based data augmentation for skin disease classification: Impact across original medical datasets to fully synthetic images. In *MICCAI workshop*, 2023.

[2] Chitwan Saharia amd William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022.

[3] Omri Avrahami, Thomas Hayes, Oran Gafni, Sonal Gupta, Yaniv Taigman, Devi Parikh, Dani Lischinski, Ohad Fried, and Xi Yin. Spatext: Spatio-textual representation for controllable image generation. In *CVPR*, 2023.

[4] Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. Synthetic data from diffusion models improves imagenet classification. *TMLR*, 2023.

[5] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *ICLR*, 2019.

[6] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. COCO-Stuff: Thing and stuff classes in context. In *CVPR*, 2018.

[7] Marlène Careil, Jakob Verbeek, and Stéphane Lathuilière. Few-shot semantic image synthesis with class affinity transfer. In *CVPR*, 2023.

[8] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *PAMI*, 40(4):834–848, 2018.

[9] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *CVPR*, 2020.

[10] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022.

[11] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The Cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016.

[12] Guillaume Couairon, Marlène Careil, Matthieu Cord, Stéphane Lathuilière, and Jakob Verbeek. Zero-shot spatial layout conditioning for text-to-image diffusion models. In *ICCV*, 2023.

[13] Prafulla Dhariwal and Alex Nichol. Diffusion models beat GANs on image synthesis. In *NeurIPS*, 2021.

[14] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using Real NVP. In *ICLR*, 2017.

[15] Weichen Fan, Jinghuan Chen, Jiabin Ma, Jun Hou, and Shuai Yi. Styleflow for content-fixed image to image translation. *arXiv*, 2207.01909, 2022.

[16] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. In *ECCV*, 2022.

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

[18] Reyhane Askari Hemmat, Mohammad Pezeshki, Florian Bordes, Michal Drozdzal, and Adriana Romero-Soriano. Feedback-guided data synthesis for imbalanced classification. *arXiv*, 2310.00158, 2023.

[19] Dan Hendrycks and Kevin Gimpel. Bridging nonlinearities and stochastic regularizers with Gaussian error linear units. *arXiv*, 1606.08415, 2016.

[20] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *NeurIPS*, 2017.

[21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

[22] Xun Huang, Arun Mallya, Ting-Chun Wang, and Ming-Yu Liu. Multimodal conditional image synthesis with product-of-experts GANs. In *ECCV*, 2022.

[23] Drew A. Hudson and C. Lawrence Zitnick. Generative adversarial transformers. In *ICML*, 2021.

[24] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017.

[25] Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up GANs for text-to-image synthesis. In *CVPR*, 2023.

[26] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019.

[27] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *CVPR*, 2020.

[28] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. In *NeurIPS*, 2021.

[29] Diederik Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1×1 convolutions. In *NeurIPS*, 2018.

[30] Diederik Kingma and Max Welling. Auto-encoding variational Bayes. In *ICLR*, 2014.

[31] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv preprint*, 2023.

[32] Nupur Kumari, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Ensembling off-the-shelf models for GAN training. In *CVPR*, 2022.

[33] Kwonjoon Lee, Huiwen Chang, Lu Jiang, Han Zhang, Zhuowen Tu, and Ce Liu. ViTGAN: Training GANs with vision transformers. In *ICLR*, 2022.

[34] Shijie Li, Ming-Ming Cheng, and Juergen Gall. Dual pyramid generative adversarial networks for semantic image synthesis. In *BMVC*, 2022.

[35] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017.

[36] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021.

[37] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. In *CVPR*, 2022.

[38] Thomas Lucas, Konstantin Shmelkov, Karteek Alahari, Cordelia Schmid, and Jakob Verbeek. Adaptive density estimation for generative models. In *NeurIPS*, 2019.

[39] Qi Mao, Hsin-Ying Lee, Hung-Yu Tseng, Siwei Ma, and Ming-Hsuan Yang. Mode seeking generative adversarial networks for diverse image synthesis. In *CVPR*, 2019.

[40] Stanislav Morozov, Andrey Voynov, and Artem Babenko. On self-supervised image representations for gan evaluation. In *ICLR*, 2021.

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

[42] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-GAN: Training generative neural samplers using variational divergence minimization. In *NeurIPS*, 2016.

[43] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv*, 1807.03748, 2019.

[44] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, 2019.

[45] Taesung Park, Alexei A. Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive learning for unpaired image-to-image translation. In *ECCV*, 2020.

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

[47] Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with VQ-VAE-2. In *NeurIPS*, 2019.

[48] Stephan R. Richter, Hassan Abu AlHaija, and Vladlen Koltun. Enhancing photorealism enhancement. *IEEE TPAMI*, 45(2):1700–1715, 2022.

[49] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[50] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[51] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *NeurIPS*, 2016.

[52] Axel Sauer, Kashyap Chitta, Jens Müller, and Andreas Geiger. Projected GANs converge faster. In *NeurIPS*, 2021.

[53] Edgar Schönfeld, Vadim Sushko, Dan Zhang, Juergen Gall, Bernt Schiele, and Anna Khoreva. You only need adversarial supervision for semantic image synthesis. In *ICLR*, 2021.

[54] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[55] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathan Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016.

[56] Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *ICML*, 2019.

[57] Hao Tang, Guolei Sun, Nicu Sebe, and Luc van Gool. Edge guided GANs with multi-scale contrastive learning for semantic image synthesis. *PAMI*, 45:14435–14452, 2023.

[58] Arash Vahdat and Jan Kautz. NVAE: A deep hierarchical variational autoencoder. In *NeurIPS*, 2020.

[59] Tengfei Wang, Ting Zhang, Bo Zhang, Hao Ouyang, Dong Chen, Qifeng Chen, and Fang Wen. Pretraining is all you need for image-to-image translation. *arXiv*, 2205.12952, 2022.

[60] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional GANs. In *CVPR*, 2018.

[61] Weilun Wang, Jianmin Bao, Wengang Zhou, Dongdong Chen, Dong Chen, Lu Yuan, and Houqiang Li. Semantic image synthesis via diffusion models. *arXiv preprint*, 2207.00050, 2022.

[62] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *ECCV*, 2018.

[63] Han Xue, Zhiwu Huang, Qianru Sun, Li Song, and Wenjun Zhang. Freestyle layout-to-image synthesis. In *CVPR*, 2023.

[64] Dingdong Yang, Seunghoon Hong, Yunseok Jang, Tianchen Zhao, and Honglak Lee. Diversity-sensitive conditional generative adversarial networks. In *ICLR*, 2019.

[65] Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. Dilated residual networks. In *CVPR*, 2017.

[66] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *ICML*, 2019.

[67] R. Zhang, P. Isola, A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

[68] Hanqing Zhao, Dianmo Sheng, Jianmin Bao, Dongdong Chen, Dong Chen, Fang Wen, Lu Yuan, Ce Liu, Wenbo Zhou, Qi Chu, Weiming Zhang, and Nenghai Yu. X-Paste: Revisiting scalable copy-paste for instance segmentation using CLIP and StableDiffusion. In *ICML*, 2023.

[69] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ADE20K dataset. In *CVPR*, 2017.