Few-shot Semantic Image Synthesis with Class Affinity Transfer

Marlène Careil[1,2] Jakob Verbeek[2] Stéphane Lathuilière[1]
[1]LTCI, Télécom Paris, IP Paris [2]Meta AI

Abstract

Semantic image synthesis aims to generate photo realistic images given a semantic segmentation map. Despite much recent progress, training them still requires large datasets of images annotated with per-pixel label maps that are extremely tedious to obtain. To alleviate the high annotation cost, we propose a transfer method that leverages a model trained on a large source dataset to improve the learning ability on small target datasets via estimated pairwise relations between source and target classes. The class affinity matrix is introduced as a first layer to the source model to make it compatible with the target label maps, and the source model is then further finetuned for the target domain. To estimate the class affinities we consider different approaches to leverage prior knowledge: semantic segmentation on the source domain, textual label embeddings, and self-supervised vision features. We apply our approach to GAN-based and diffusion-based architectures for semantic synthesis. Our experiments show that the different ways to estimate class affinity can be effectively combined, and that our approach significantly improves over existing state-of-the-art transfer approaches for generative image models.

1. Introduction

Image synthesis with deep generative models has made remarkable progress in the last decade with the introduction of GANs [11], VAEs [17], and diffusion models [14]. Generated images can be conditioned on diverse types of inputs, such as class labels [3, 18], text [10, 25, 27], bounding boxes [35], or seed images [5]. In semantic image synthesis, the generation is conditioned on a semantic map that indicates the desired class label for every pixel. This task has been thoroughly explored with models such as SPADE [23] and OASIS [33], capable of generating high-quality and diverse images on complex datasets such as ADE20K [43] and COCO-Stuff [4]. However, these approaches heavily rely on the availability of large datasets with tens to hundreds of thousands of images annotated with pixel-precise label maps that are extremely costly to acquire. For the Cityscapes dataset [7], *e.g.*, on average more than 1.5h per image was required for annotation and quality control.

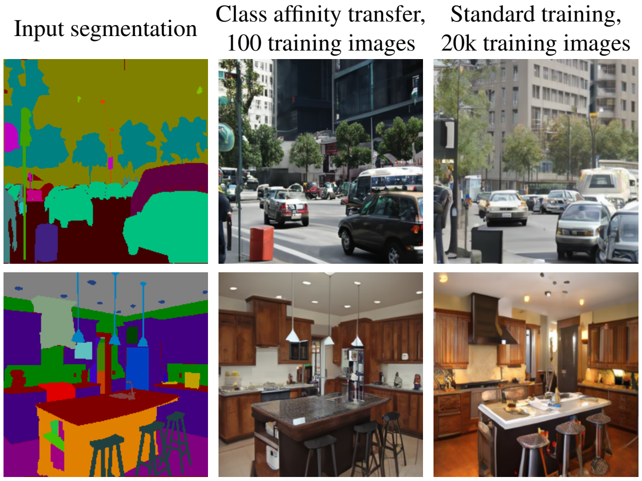
Input segmentation
Class affinity transfer, 100 training images
Standard training, 20k training images

Figure 1. **Can we train a semantic image synthesis model from only 100 images?** Our diffusion-based transfer results using training set of 100 ADE20K images (2$^{\text{nd}}$ col.) compared to the same model trained from scratch on full dataset (20k images, 3$^{\text{rd}}$ col.).

High annotation costs can be a barrier to deployment of machine learning models in practice, and motivates the development of transfer learning strategies to alleviate the annotation requirements. These techniques allow training models on small target datasets via the use of models pretrained on a source dataset with many available annotations. Transfer learning has been widely studied for classification tasks such as object recognition [2, 16, 26], but received much less attention in the case of generation tasks. This task has been considered for unconditional and class-conditional generative models [19, 21, 22, 38, 39, 41], but to the best of our knowledge few-shot transfer learning has not yet been explored in the setting of semantic image synthesis.

We introduce **CAT**, a finetuning procedure that models **C**lass **A**ffinity to **T**ransfer knowledge from pre-trained semantic image synthesis models. Our method takes advantage of prior knowledge to establish pairwise relations between source and target classes, and encodes them in a class affinity matrix. This solution considerably eases learning when few instances of the target classes are available at training time. The affinity matrix is prepended to the source model to make it compatible with the label space of the target domain. The model can then be further finetuned using the available data for the target domain. To illustrate the generality of the proposed approach, we integrate our transfer learning strategy in state-of-the-art adversarial and diffusion models. We explore different ways to extract similarities between source and target classes, using semantic segmentation models for the source data, self-supervised vision features, and text-based class embeddings.

We conduct extensive experiments on the ADE20K, COCO-Stuff, and Cityscapes datasets, using target datasets with sizes ranging from as little as 25 up to 400 images. Our experiments show that our approach significantly improves over state-of-the-art transfer methods. As illustrated in Figure 1, our approach allows realistic synthesis from no more than 100 target images, and achieves image quality close to standard training on the full target datasets. Moreover, unlike previous transfer methods, our approach also enables non-trivial training-free transfer results, where we only prepend the class affinity matrix to the source model, without further finetuning it.

In summary, our contributions are the following:

- We introduce Class Affinity Transfer (CAT), the first transfer method for semantic image synthesis for small target datasets, and explore different methods to define class affinity, based on semantic segmentation, self-supervised features, and text-based similarity.
- We integrate our approach in state-of-the-art adversarial and diffusion based semantic synthesis models.
- We obtain excellent experimental transfer results, improving over existing state-of-the-art approaches.

2. Related work

Semantic image synthesis with GANs. There has been significant interest in adversarial approaches for semantic image synthesis, see *e.g*. [1, 15, 23, 33]. These approaches employ a conditional generator and a discriminator that assesses both image quality and consistency with the input segmentation maps. One of the first models proposed was Pix2Pix [15], which uses a U-Net [31] generator along with a patch-based discriminator. SPADE [23] employs a different generator with spatially adaptive normalization layers modulating feature maps through labels. Lab2Pix-V2 [45] introduces special modules in the generator for extracting meaningful information from labels. OASIS [33] overcomes the need of perceptual loss with the introduction of a U-Net discriminator which produces per-pixel classification scores. This approach obtains state-of-the-art results on the task of semantic image synthesis, and we build upon it in our GAN-based experiments.

Semantic synthesis with diffusion-based models. Recently, diffusion models [14] have emerged as a promising solution capable of synthesizing images with a quality that surpasses GANs [8, 30]. Generation is formulated as an iterative denoising process and a likelihood-based loss is used as training objective which makes training more stable and scalable to large datasets. A few works address semantic image synthesis with diffusion models. In [37], SPADE blocks are included in the U-Net used in the diffusion model to improve the semantic consistency of the generated images. PITI [36] builds upon GLIDE [20], a text-conditioned diffusion model pre-trained on hundreds of millions of image-text pairs. The text encoder is then replaced by one that takes semantic segmentation maps as input. Different from our work, PITI focuses on transferring a text-based model to a semantic synthesis model, and still requires a large training set to train the semantic map encoder network from scratch (20k to 110k training images in their experiments on ADE20K and COCO-Stuff). In contrast, we address transfer from existing semantic synthesis models, in scenarios where only few target images are available: from 25 to 400 in our experiments.

Transfer learning for generative models. Compared to transfer learning for discriminative problems, transfer for generative models received much less attention. In the seminal work [39], a pre-trained unconditional GAN is finetuned either for conditional or unconditional generation in limited data regimes. Several works show that this approach can be improved by finetuning only part of the network parameters [19, 21]. Another strategy consists in adding and learning a limited number of additional parameters [29, 38, 41]. For instance, Zhao *et al*. [41] apply affine modulations to the frozen parameters of the pre-trained model. In MineGAN [38], a small "miner" network is introduced to warp the distribution of the latent variable to better fit the target distribution. In [12], they perform few-shot image generation by using a local fusion module in the encoder space. Other approaches investigate the use of style transfer for few-shot GAN training [22, 42].

A few works specifically consider transfer for conditional GANs. Shahbazi *et al*. [34] propagate information from old classes to new classes through the use of batch normalization. Their work, however, considers class conditional generation, which is different from our work where we condition on semantic segmentation maps. Endo *et al*. [9] use an unconditional StyleGAN2 model to generate synthetic images with pseudo labels. They do this by learning a nearest-centroid classifier in the GAN latent space from real images with corresponding label maps, where a PSP encoder [28] is used to obtain the latents of real images.

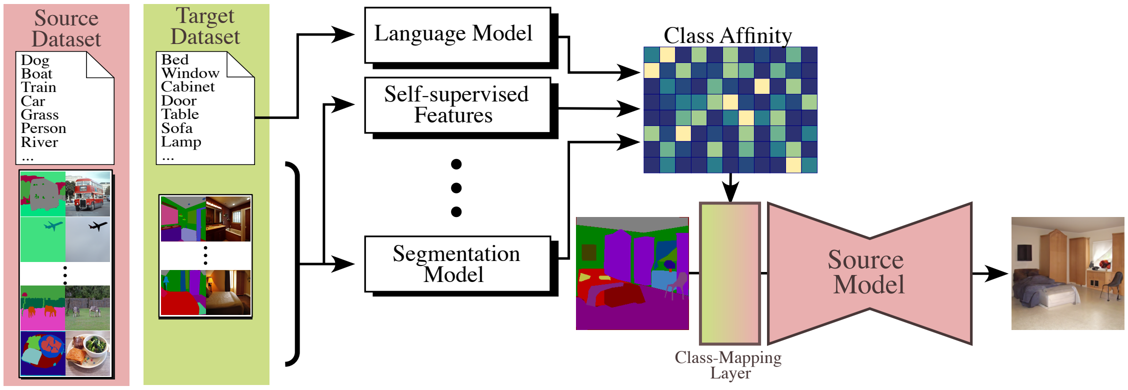

Figure 2. Overview of our class affinity transfer (CAT) approach for semantic image synthesis. The class affinity matrix is used to align the source model with the target label space, and is then further finetuned using the target images and corresponding segmentations.

Although interesting, this approach hinges on the availability of a strong unconditional generative model and the ability of faithful latent inference, which is possible for datasets with limited diversity such as human faces, but which is extremely challenging for complex datasets such as ADE20K and COCO-Stuff that we consider in our experiments.

There are very few works on model transfer for diffusion models. In addition to PITI, which considers semantic image synthesis as discussed above, Ruiz *et al.* [32] consider the problem of instance-driven generation. They finetune a pretrained text-based diffusion model on a handful of images of a particular object in different contexts. The finetuned model is then used to generate images with the same object in other environments described by a textual prompt.

3. Class affinity transfer

We aim at adapting a semantic image synthesis model pre-trained on a large source dataset to a small target dataset. We assume a source dataset composed of N RGB images $\mathbf{X}_n \in \mathbb{R}^{H\times W\times 3}, n \in \{1,\dots,N\}$ and their corresponding segmentation maps $\mathbf{S}_n \in \{0,1\}^{H\times W\times C_\mathcal{S}}$, represented with one-hot encoding across $C_\mathcal{S}$ source classes. Similarly, we consider a target dataset composed of M images $\mathbf{X}^\mathcal{T}_m \in \mathbb{R}^{H\times W\times 3}, m \in \{1,\dots,M\}$ and their corresponding segmentation maps $\mathbf{S}^\mathcal{T}_m \in \{0,1\}^{H\times W\times C_\mathcal{T}}$ with $C_\mathcal{T}$ target classes. Note that, the number of classes is different between the source and target datasets, and that no correspondence between them is given.

Our class affinity transfer (CAT) approach is based on an affinity matrix $\mathbf{A} \in \mathbb{R}^{C_\mathcal{S}\times C_\mathcal{T}}$ that maps between the classes of the source and target datasets. We use the affinity matrix to parameterize a linear layer which we prepend to the source model, making the source model compatible with one-hot input label maps of the target domain.

Rather than initializing the affinity matrix at random, we leverage different forms of prior knowledge to estimate the affinity matrix, which enables the source model to adapt significantly better to the target domain. The model can then be further finetuned using training images and segmentation maps from the target domain. In addition, our approach also allows for "training-free" transfer mode where we fully rely on the class affinity transfer matrix, further finetuning on the target dataset.

Below, we describe different approaches to estimate the class affinity matrix in §3.1. Then, we show how our approach can be incorporated into a state-of-the-art GAN and diffusion based architectures in §3.2 and §3.3, respectively.

3.1. Estimating the class affinity matrix

We consider three different ways to estimate the class affinity matrix $\mathbf{A}$ between the source and target classes, leveraging different forms of prior knowledge.

Supervised semantic segmentation networks. Here we employ a pre-trained segmentation network trained on the source dataset to extract mappings between source and target classes. First, we segment the target images across the source classes using the segmentation network. Next, we use segmentation maps of the target images and count how many pixels of each target class are classified as each source class to establish the class affinity matrix.

More formally, we denote by $\bar{\mathbf{S}}_m \in \{0,1\}^{H\times W\times C_\mathcal{S}}$ the output of the segmentation network for a target image $\mathbf{X}^\mathcal{T}_m$, and use subscripts i, j to denote pixel locations, and superscripts k and l to index across target and source classes. The affinity matrix $\mathbf{A}$ is computed as the confusion matrix between source and target classes:

$$\mathbf{A}_{k,l} \propto \sum_{m=1}^{M} \sum_{i,j} [\mathbf{S}^\mathcal{T}_m]^k_{i,j} \cdot [\bar{\mathbf{S}}_m]^l_{i,j}. \qquad (1)$$

The matrix is normalized so that for each target class the affinities w.r.t. all source classes sum to one, *i.e.* $\sum_{l=1}^{C_{\mathcal{S}}} \mathbf{A}_{k,l} = 1$. In this manner, prepending the affinity matrix as a linear layer to the network leaves the scale of the input comparable with inputs from the source dataset. In our experiments, we use UperNet [40] or DeeplabV2 [6] as pretrained segmentation networks.

Self-supervised image features. To alleviate the requirement of training a dedicated segmentation network on the source dataset, we explore self-supervised learning (SSL) to extract features from image patches using iBOT [44]. Using the corresponding segmentation maps, we represent each class in the source and target dataset using a "prototype" which is obtained as the weighted average of the features of patches that belong to that class. The features of each patch are weighted proportionally to the number of pixels with a given label in the patch. We denote these prototypes as $\boldsymbol{f}_l^{\mathcal{S}} \in \mathbb{R}^D$ and $\boldsymbol{f}_k^{\mathcal{T}} \in \mathbb{R}^D$ where D is the embedding dimension, for source and target classes respectively. We then compute the class affinities using cosine similarity between the prototypes:

$$\mathbf{A}_{k,l} \propto \cos(\boldsymbol{f}_k^{\mathcal{T}}, \boldsymbol{f}_l^{\mathcal{S}}), \tag{2}$$

and similarly normalize the affinities so that for each target class they sum to one across the source classes.

Text-based class affinities. The previous approaches estimate class affinities using source and target images with corresponding segmentation maps. Here we consider an alternative that does not require any labeled images, and instead relies on the class names to establish affinities. To this end, we use a pre-trained CLIP [24] text encoder to embed the names of the source and target classes as $\boldsymbol{f}_l^{\mathcal{S}} \in \mathbb{R}^D$ and $\boldsymbol{f}_k^{\mathcal{T}} \in \mathbb{R}^D$. Similar to Eq. (2), we obtain the affinities as normalized cosine similarities over the text embeddings.

Combination via majority voting. To take advantage of the three different methods to estimate the class affinities, we introduce an aggregation scheme that combines the affinity matrices obtained with all the previous methods. While the different estimations of **A** could be combined via simple averaging, we obtain better performance with a binary majority voting scheme. If, for a given target class, at least two of the three affinity matrices agree on source class with highest affinity, then the target class is associated with the corresponding source class. If the three affinity matrices disagree, we take the source class provided by the method with the lowest initial, *i.e.* "training free", FID on the training target set.

3.2. Few-shot transfer with GAN

We integrate our class affinity transfer approach with the state-of-the-art OASIS semantic image synthesis model [33]. It consists of a convolutional generator with SPADE blocks [23] to condition on segmentation maps, and a U-Net [31] discriminator to label pixels of real images with the corresponding class, and generated pixels as "fake". Based on initial experiments, we introduce several modifications to both generator and discriminator to improve transfer to small target datasets.

Architecture. First, we prepend the class affinity matrix to the SPADE blocks that take the segmentation map as input to align them with the target label space. Second, rather than sharing the first convolutional layer, we use separate paths for the scale and shift parameters, but still share the parameters of the first convolutional layer. Third, we add two parallel branches from the input which bypass the class affinity matrix and the first convolutional layer. These branches take the target segmentation map as input and project the latter to the feature space of the first pretrained convolution output in each of the SPADE block. We then sum these residual outputs to the main branch. The weights of the parallel branch are initialized with zeros to prevent negative impact early in training. The motivation behind this design choice is to enable the generator to better learn how to synthesize new target classes which could not be explained by a linear combination of source classes.

In the discriminator, we replace the last convolutional layer (which outputs per-pixel classification scores) by a randomly initialized layer with an output channel size corresponding to the number of classes in the target dataset. Alternatively, we experimented with a linear layer initialized with our affinity matrix added on top of the discriminator to map the source and target classes, but this approach did not improve performance.

Finetuning. Similar to [19], we found that freezing the first layers of the discriminator is beneficial when finetuning the source model for the target datasets. Regarding the generator, we proceed in two stages. In the first stage, we fix most generator parameters and only finetune the class affinity matrix, the following first convolution layer, and the residual branch in each SPADE block. In the second stage, we finetune all the layers in the generator. The losses used during finetuning are the same as during pretraining, *i.e.* we use an adversarial loss as well as the LabelMix [33] regularization loss.

3.3. Few-shot transfer with diffusion model

Architecture. For our diffusion-based experiments, we adopt the PITI [36]. It is a modified version of GLIDE [20], a text-conditioned diffusion model that generates the image via iterative denoising using a U-Net. GLIDE consists of two text-conditional networks: the first generates a 64×64

image; the second upsamples the image to 256×256 resolution. In PITI, the text encoder of both networks is replaced by a semantic map encoder with a transformer architecture.

To be compatible with our class affinity transfer approach, we modify PITI to be conditioned on one-hot label maps rather than RGB label maps. We do this by factoring the class-to-RGB mapping into the weights of the first layer of the encoder network. Similarly to the GAN-based model, we use the class affinity matrix $\mathbf{A}$ to parameterize a linear layer which we prepend to the semantic image encoder. To allow further adaptation to the target task, we take inspiration from [16], and introduce trainable extra parameters in the transformer encoder, referred to as "prompts", which can be seen as additional patch embeddings in input of each attention layer. The prompts are randomly initialized. For more details see the supplementary material.

finetuning. To finetune PITI, we freeze the decoder layers, *i.e.* the U-Net model, and only train part of the segmentation encoder. We fix all the weights in the encoder transformer, and only train the last ResNet block and the prompts of the encoder. We employ the training loss used in GLIDE, and finetune both the low resolution model as well as the conditional upsampling model, as in [36].

4. Experiments

4.1. Experimental setup

Datasets. In order to make our research results comparable to earlier work on semantic image synthesis, we employ ADE20K [43] (20k images and 151 classes) and COCO-Stuff [4] (110k images and 183 classes) as source datasets. As target datasets, we use subsets of ADE20K and COCO-Stuff, as well as the Cityscapes [7] dataset consisting of 3k images and 35 classes. To avoid training our models on personal data, we use the version of the Cityscapes dataset with blurred faces and license plates, while for COCO-Stuff and ADE20K we applied a face blurring pipeline ourselves. Following [23, 33], we train models at 256×256 resolutions for ADE20K and COCO-Stuff, and 256×512 for Cityscapes. For PITI [36], we also use a resolution of 256×256 for Cityscapes since the positional embeddings in PITI are pretrained at 256×256 .

We sample subsets as target datasets to evaluate the different methods in few-shot regimes. To ensure that all the target classes are well represented in the subsets, we use a specific sampling procedure. We take an initial random image and then iteratively select subsequent images such that the KL-divergence between the uniform distribution and the empirical class distribution is minimized. The empirical class distribution is obtained by counting how many pixels of each class are present in each segmentation map, and normalizing the histogram to sum to one. Unless otherwise indicated, we use target subsets of 100 images in all our experiments. The impact of the target set size is discussed in Section 4.3 where we perform experiments with subsets ranging in size from 25 to 400 images.

Evaluation metrics. We report both FID [13] and mIoU metrics as in [15, 23, 33]. FID captures both image quality and diversity, while mIoU assesses the semantic correspondence with the input segmentation maps by using a segmentation network to label generated images. We use the same segmentation networks as in [33].

Baselines. For OASIS [33], the most basic comparison is to training the model from scratch, without any transfer. To the best of our knowledge, we are the first to propose a transfer method specifically developed for semantic image synthesis. Therefore, to evaluate our model, we compare to existing transfer learning works developed for unconditional and class-conditioned GANs by adapting them for semantic image synthesis. We compare to TransferGAN [39] by finetuning all the layers of the source generator and discriminator to adapt to the target dataset. We also compare to Freeze-D [19], which finetunes both generator and discriminator, while freezing the layers of the discriminator closest to the input image. Based on the ablations in [19], we consider freezing the first up to the ninth layers of the discriminator. BSA [21] finetunes only the batch normalization (BN) parameters. In OASIS, the BN parameters are computed from the label maps through SPADE blocks. Therefore, for BSA we only finetune the first layer of the SPADE blocks and freeze all the other weights. MineGAN [38] freezes the source generator and adds a small MLP mapping network that transforms the latent vector, while finetuning the source discriminator to the target dataset. To adapt it to semantic image synthesis, we also learn the class embedding layer of the generator for the target dataset. We follow the two-stage training approach of MineGAN, where in the second training stage we finetune the entire generator and discriminator networks. We also test cGANTransfer [34], by finetuning scale and shift parameters in SPADE blocks, projecting each target class as a trainable linear combinations of source classes. Unlike CAT, these linear combinations are initialized randomly. We also add trainable residual layers after the first convolution projecting label maps and train with the ℓ_1 and ℓ_2 regularization losses of [34].

For PITI [36], we compare our method to finetuning from a pretrained GLIDE [20] model. In this baseline, we train the encoder from scratch to map segmentation maps to the latent space of GLIDE, as in [36]. We also consider a baseline where we finetune all layers in a pretrained PITI, by re-initializing the first encoder layer that takes as input the segmentation map.

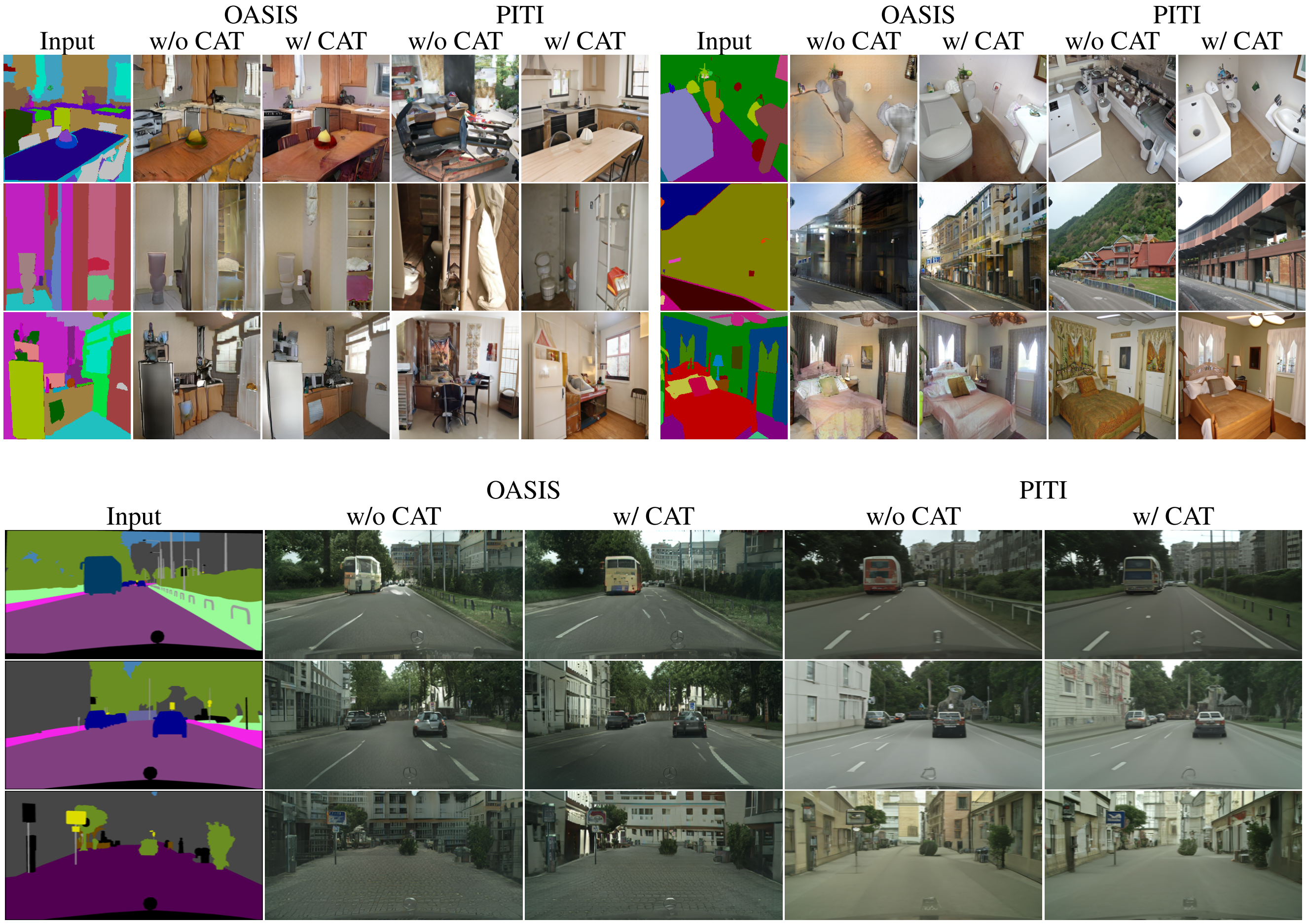

Figure 3. Samples from models trained with 100 target images. Transfer from ADE to COCO (top left), COCO to ADE (top right), and from COCO to Cityscapes (bottom). Class affinity matrix initialized randomly (w/o CAT) or with combination method (w/ CAT).

	Affinity matrix initialization	**COCO→ADE**		**ADE→COCO**	
		↓FID	**↑mIoU**	**↓FID**	**↑mIoU**
OASIS	Random	54.0	30.0	82.9	15.9
	Text-based	41.1	30.4	55.2	17.3
	Segmentation	42.0	30.8	58.2	12.9
	Self-supervised	41.3	29.8	57.9	15.4
	Combination	**40.9**	**31.4**	**53.7**	**17.4**
PITI	Random	57.1	11.6	83.7	0.8
	Text-based	40.9	20.2	47.4	7.1
	Segmentation	41.1	22.0	52.5	5.3
	Self-supervised	41.9	21.2	50.9	5.6
	Combination	**40.7**	**22.3**	**46.8**	**7.5**

Table 1. Comparison of different class affinity estimation methods and their combination to randomly initializing the affinity matrix. Results after finetuning.

4.2. Main affinity estimation results

Quantitative evaluation. In our first experiment we evaluate the performance of the class affinity estimation from class label embeddings, semantic segmentation, self-supervised features, and their combination, when transferring from COCO-Stuff to ADE20K and vice-versa. We also compare to randomly initializing the affinity matrix. In all cases, we initialize the other weights from the source model, and finetune all weights on the target data.

The results in Table 1 show consistent gains over the random initialization baseline across the board. The text and segmentation based affinities perform better than the self-supervised features, and the combination of all three yields the best results on both transfer problems and metrics. In particular, when comparing random initialization and the combined class affinities for OASIS, we improve FID from 54.0 to 40.9 and mIoU from 30.0 to 31.4 when transferring from COCO to ADE, and improve FID from 82.9 to 53.7 and mIoU from 15.9 to 17.4 in the reverse tranfer direction.

	Method	COCO→ADE		ADE→COCO		ADE→Cityscapes		COCO→Cityscapes	
		↓FID	↑mIoU	↓FID	↑mIoU	↓FID	↑mIoU	↓FID	↑mIoU
OASIS	From scratch	145.9	13.6	153.4	7.1	136.5	37.1	137.0	37.2
	TransferGAN [39]	85.1	20.4	120.5	10.2	56.2	61.5	51.5	63.6
	FreezeD [19]	66.3	25.9	102.4	13.8	57.1	62.7	49.8	66.5
	MineGAN [38]	82.2	21.0	110.2	11.5	57.8	62.2	52.6	65.4
	BSA [21]	70.1	25.9	94.2	12.8	76.3	51.6	65.7	59.1
	cGAN-Transfer [34]	64.9	26.2	89.8	15.0	63.6	61.7	57.3	58.9
	CAT (ours)	**40.9**	**31.4**	**53.7**	**17.4**	**51.4**	**66.1**	**47.0**	**68.1**
PITI	From GLIDE [37]	59.8	2.0	104.9	0.3	74.5	9.5	74.5	9.5
	Finetune all	56.8	14.2	83.7	0.1	86.1	17.2	70.8	36.7
	CAT (ours)	**40.7**	**22.3**	**46.8**	**7.5**	**62.7**	**27.3**	**54.7**	**39.9**

Table 2. Comparison with state-of-the-art transfer methods, using target datasets of 100 images.

For PITI, we improve FID from 57.1 to 40.7 and mIoU from 11.6 to 22.3 when transferring from COCO to ADE, and improve FID from 83.7 to 46.8 and mIoU from 0.8 to 7.5 in the opposite direction.

While the FID values for the diffusion-based PITI model are comparable or better than those obtained using OASIS, we noticed that mIoU values for the diffusion-based model are worse. This trend was already observed on the source dataset: PITI trained on the full COCO dataset has an mIoU of 34.4, while the mIoU on OASIS is 44.1. This gap widens when training on the full ADE dataset, where PITI has an mIoU of 26 compared to 48.8 for OASIS.

Qualitative results. In Figure 3, we show samples synthesized from PITI and OASIS models with three different types of transfer, from ADE to COCO on top left, from COCO to ADE on top right and from COCO to Cityscapes on the bottom, finetuning with 100 target images with and without class affinity transfer (CAT). Both for PITI and OASIS, training with CAT leads to synthesized images with sharper details and better recognizable objects. For instance, the sink and bathtub in images of the first row in the top right of the figure are of better quality and more realistic when trained with CAT. Furthermore, when transferring to the challenging COCO dataset containing 183 classes, we notice that without CAT, PITI fails to synthesize realistic images adhering to the label maps, whereas CAT can synthesize images of better quality coherent with label maps.

Comparison with the state of the art. We compare to the state-of-the-art transfer methods for generative models. We consider four pairs of target-source datasets, by taking source models trained either on COCO-Stuff or ADE20K datasets, and finetuning them on target datasets of 100 images taken from Cityscapes, ADE20K and COCO-Stuff.

From the results in Table 2 using the OASIS architecture, we observe a significant improvements with CAT. We improve FID from 64.9 to 40.9 and mIoU from 26.2 to 31.4 COCO→ADE, and improve FID from 89.8 to 53.7 and mIoU from 15.0 to 17.4 for ADE→COCO w.r.t. the best baseline cGAN-Transfer. When transferring to Cityscapes, CAT improves FID from 56.2 (TransferGAN) to 51.4 and mIoU from 62.7 (FreezeD) to 66.1 when the source dataset is ADE, and improving FID from 49.8 to 47.0 and mIoU from 66.5 to 68.1 w.r.t. FreezeD with COCO as source.

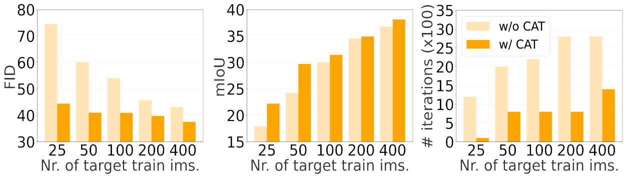

Figure 4. FID, mIoU and # training iterations for COCO→ ADE transfer using OASIS w/ and w/o CAT for different dataset sizes.

In the case of PITI, directly finetuning a pretrained GLIDE model on the target dataset ("From GLIDE") produces images with much better quality compared to OASIS trained from scratch in terms of FID. However, the model fails to generate images with strong adherence to label maps, as reflected by the poor mIoU scores. Generally, finetuning PITI trained on the source dataset on the target dataset ("finetune all") gives better results than finetuning from GLIDE. CAT significantly improves over these two baselines in all settings: with more than 15 points in FID, and more than 3 points in mIoU.

4.3. Ablation study and analysis

Impact of the target dataset sizes. We generate subsets of size ranging from 25 to 400 images on ADE and analyze the peformance of CAT using OASIS as base model. We report the evolution of FID and mIoU in Figure 4 (left and center panel, respectively). While CAT (in dark orange) demonstrates gains in both FID and mIoU in all the dataset sizes, its advantage is striking in the very low-shot setting. CAT surpasses by large margins of 30.1 and 19.1 FID points, respectively on datasets of size 25 and 50. Even if the boost

	Affinity matrix initialization	COCO→ADE ↓FID	COCO→ADE ↑mIoU	ADE→COCO ↓FID	ADE→COCO ↑mIoU
OASIS	Random	216.0	0.5	270.8	0.1
	Text-based	44.6	23.9	57.8	15.2
	Segmentation	47.0	22.8	64.5	12.1
	Self-supervised	45.5	22.7	68.1	10.2
	Combination	**43.1**	**25.1**	**56.3**	**13.8**
	Combo + finetuning	*40.9*	*31.4*	*53.7*	*17.4*
PITI	Random	96.3	5.7	98.3	0.1
	Text-based	51.6	19.2	50.8	7.1
	Segmentation	48.7	20.2	59.2	5.2
	Self-supervised	49.5	19.6	53.9	5.0
	Combination	**48.5**	**20.9**	**49.3**	**7.4**
	Combo + finetuning	*40.7*	*22.3*	*46.8*	*7.5*

Table 3. Training-free transfer results. Results obtained with additional finetuning are marked in italic.

FreezeD	2 stages	Resid.	CAT	COCO→ADE ↓FID	COCO→ADE ↑mIoU	ADE→COCO ↓FID	ADE→COCO ↑mIoU
✗	✗	✗	✗	87.2	20.7	117.3	11.0
✓	✗	✗	✗	65.7	25.8	98.6	14.8
✓	✓	✗	✗	55.9	28.6	83.4	15.2
✓	✓	✓	✗	55.2	29.4	79.9	15.7
✓	✓	✓	✓	**40.9**	**31.4**	**53.7**	**17.4**

Table 4. Ablations with adversarial OASIS architecture.

FixDec	Prompts	CAT	COCO→ADE ↓FID	COCO→ADE ↑mIoU	ADE→COCO ↓FID	ADE→COCO ↑mIoU
✗	✗	✗	56.5	13.5	85.0	0.1
✓	✗	✗	52.4	13.6	79.0	1.4
✓	✓	✗	51.1	14.1	78.9	1.3
✓	✓	✓	**40.7**	**22.3**	**46.8**	**7.5**

Table 5. Ablations with diffusion-based PITI architecture.

narrows when increasing the dataset size, we still observe a performance gain of 5.6 points of FID with 400 training images. For reference, training OASIS on the full dataset ADE (with face blurring) achieves 29.8 of FID and 48.6 of mIoU. In the right panel of Figure 4 we report the number of training iterations before convergence. We note a faster convergence with CAT for all the dataset sizes.

Training-free transfer. If we only add the class affinity matrix to the source model, without further finetuning the model on the target data, it can already be used to generate samples for the target domain. Since the class-affinity matrix is obtained with a single feed-forward pass through the target training data, or even just using the textual class embeddings, this approach can be considered "training-free" and is extremely computationally efficient.

From the results in Table 3, we notice that the affinity matrix that combines the different methods consistently obtains the best (or very close) performance in terms of FID and mIoU. No matter the choice of affinity estimation method, our training-free variant of CAT achieves performance far better than using a randomly initialized class affinity matrix. Both in terms of FID and mIoU, the training-free result is already relatively close to the results obtained with finetuning ("Combo + finetuning"). This underlines the key role of a good initialization for transfer for semantic image synthesis. Qualitative results of training-free transfer can be found in the supplementary material.

Ablations for OASIS. In Table 4, we ablate how each component contributes to the performance. We observe that freezing part of the discriminator parameters, as done in [19], improves performance. We also found it beneficial to perform finetuning in two stages: in the first stage, we fix most of our generator parameters, and only finetune the first convolution in each SPADE block that takes as input the segmentation map; in the second stage, we train all the generator parameters. We also demonstrate that adding residual convolutional layers is beneficial (*e.g.* from 83.4 to 79.9 in FID in ADE→COCO). Finally, we obtain substantial gains using our class affinity matrix for initialization.

Ablations for PITI. The ablation study in the case of the diffusion-based model is reported in Table 5. We observe better performance when fixing the decoder part, and introducing trainable prompts in the segmentation map encoder further improves FID and mIoU. Lastly, when adding our class affinity matrix, we consistently improve performance by a larger margin in all settings according to both metrics.

5. Conclusion

In this paper, we consider the problem of few-shot semantic image synthesis, where training sets consist of a few tens to a few hundreds images. To address this problem, we proposed **C**lass **A**ffinity **T**ransfer (CAT), a transfer learning approach based on estimating a class affinity matrix, using the similarities among classes in the source and target datasets. We consider four methods to establish these similarities: based on a semantic segmentation model of the source domain, using self-supervised vision features, or using text-based class label embeddings, and a combination via majority voting. The class affinity matrix is prepended as a first layer to the source model to align it with the one-hot-labels of the target domain. We integrated our approach in both an adversarial (OASIS) and a diffusion-based architecture (PITI). We conducted extensive experiments on the COCO-Stuff, ADE20K, and Cityscapes datasets, and observed excellent transfer performance. Consistently outperforming state-of-the-art transfer methods for generative models, and allowing realistic semantic image synthesis using training sets as small as 100 images.

References

[1] Samaneh Azadi, Michael Tschannen, Eric Tzeng, Sylvain Gelly, Trevor Darrell, and Mario Lucic. Semantic bottleneck scene generation. *arXiv preprint*, arxiv:1911.11357, 2019. 2

[2] Rodrigo Berriel, Stéphane Lathuilière, Moin Nabi, Tassilo Klein, Thiago Oliveira-Santos, Nicu Sebe, and Elisa Ricci. Budget-aware adapters for multi-domain learning. In *ICCV*, 2019. 1

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *ICLR*, 2019. 1

[4] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. COCO-Stuff: Thing and stuff classes in context. In *CVPR*, 2018. 1, 5

[5] Arantxa Casanova, Marlène Careil, Jakob Verbeek, Michal Drozdzal, and Adriana Romero-Soriano. Instance-conditioned GAN. In *NeurIPS*, 2021. 1

[6] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille. Semantic image segmentation with deep convolutional nets and fully connected CRFs. In *ICLR*, 2015. 4

[7] 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. 1, 5

[8] Prafulla Dhariwal and Alex Nichol. Diffusion models beat GANs on image synthesis. In *NeurIPS*, 2021. 2

[9] Yuki Endo and Yoshihiro Kanamori. Few-shot semantic image synthesis using StyleGAN prior. *arXiv preprint*, arXiv:2103.14877, 2021. 2

[10] 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. 1

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NeurIPS*, 2014. 1

[12] Zheng Gu, Wenbin Li, Jing Huo, Lei Wang, and Yang Gao. LoFGAN: Fusing local representations for few-shot image generation. In *ICCV*, 2021. 2

[13] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *NeurIPS*, 2017. 5

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 1, 2

[15] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 2, 5

[16] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, 2022. 1, 5

[17] D. Kingma and M. Welling. Auto-encoding variational Bayes. In *ICLR*, 2014. 1

[18] M. Mirza and S. Osindero. Conditional generative adversarial nets. In *NIPS deep learning workshop*, 2014. 1

[19] Sangwoo Mo, Minsu Cho, and Jinwoo Shin. Freeze the discriminator: a simple baseline for fine-tuning GANs. In *CVPR AI for Content Creation Workshop*, 2020. 1, 2, 4, 5, 7, 8

[20] 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. 2, 4, 5

[21] Atsuhiro Noguchi and Tatsuya Harada. Image generation from small datasets via batch statistics adaptation. In *ICCV*, 2019. 1, 2, 5, 7

[22] Utkarsh Ojha, Yijun Li, Jingwan Lu, Alexei A. Efros, Yong Jae Lee, Eli Shechtman, and Richard Zhang. Few-shot image generation via cross-domain correspondence. In *CVPR*, 2021. 1, 2

[23] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, 2019. 1, 2, 4, 5

[24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 4

[25] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. Hierarchical text-conditionalimage generation with CLIP latents. *arXiv preprint*, arXiv:2204.06125, 2022. 1

[26] S.-A. Rebuffi, H. Bilen, and A. Vedaldi. Learning multiple visual domains with residual adapters. In *NeurIPS*, 2017. 1

[27] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. In *ICML*, 2016. 1

[28] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a StyleGAN encoder for image-to-image translation. In *CVPR*, 2021. 2

[29] Esther Robb, Wen-Sheng Chu, Abhishek Kumar, and Jia-Bin Huang. Few-shot adaptation of generative adversarial networks. *arXiv preprint*, 2020. 2

[30] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2

[31] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, 2015. 2, 4

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

[33] 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. 1, 2, 4, 5

[34] Mohamad Shahbazi, Zhiwu Huang, Danda Pani Paudel, Ajad Chhatkuli, and Luc Van Gool. Efficient conditional

GAN transfer with knowledge propagation across classes. In *CVPR*, 2021. 2, 5, 7

[35] Wei Sun and Tianfu Wu. Image synthesis from reconfigurable layout and style. In *ICCV*, 2019. 1

[36] 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 preprint*, arXiv:2205.12952, 2022. 2, 4, 5

[37] Weilun Wang, Jianmin Bao, Wengang Zhou, Dongdong Chen, Dong Chen, Lu Yuan, and Houqiang Li. Semantic image synthesis via diffusion models. *arXiv preprint arXiv:2207.00050*, 2022. 2, 7

[38] Yaxing Wang, Abel Gonzalez-Garcia, David Berga, Luis Herranz, Fahad Shahbaz Khan, and Joost van de Weijer. MineGAN: effective knowledge transfer from GANs to target domains with few images. In *CVPR*, 2020. 1, 2, 5, 7

[39] Yaxing Wang, Chenshen Wu, Luis Herranz, Joost van de Weijer, Abel Gonzalez-Garcia, and Bogdan Raducanu. Transferring GANs: generating images from limited data. In *ECCV*, 2018. 1, 2, 5, 7

[40] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *ECCV*, 2018. 4

[41] Miaoyun Zhao, Yulai Cong, and Lawrence Carin. On leveraging pretrained GANs for generation with limited data. In *ICML*, 2020. 1, 2

[42] Yunqing Zhao, Henghui Ding, Houjing Huang, and Ngai-Man Cheung. A closer look at few-shot image generation. In *CVPR*, 2022. 2

[43] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ADE20K dataset. In *CVPR*, 2017. 1, 5

[44] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. iBOT: Image BERT pretraining with online tokenizer. In *ICLR*, 2021. 4

[45] Junchen Zhu, Lianli Gao, Jingkuan Song, Yuan-Fang Li, Feng Zheng, Xuelong Li, and Heng Tao Shen. Label-guided generative adversarial network for realistic image synthesis. *PAMI*, 45(3):3311–3328, 2023. 2