LABEL-EFFICIENT SEMANTIC SEGMENTATION WITH DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Denoising diffusion probabilistic models have recently received much research attention since they outperform alternative approaches, such as GANs, and currently provide state-of-the-art generative performance. The superior performance of diffusion models has made them an appealing tool in several applications, including inpainting, super-resolution, and semantic editing. In this paper, we demonstrate that diffusion models can also serve as an instrument for semantic segmentation, especially in the setup when labeled data is scarce. In particular, for several pretrained diffusion models, we investigate the intermediate activations from the networks that perform the Markov step of the reverse diffusion process. We show that these activations effectively capture the semantic information from an input image and appear to be excellent pixel-level representations for the segmentation problem. Based on these observations, we describe a simple segmentation method, which can work even if only a few training images are provided. Our approach significantly outperforms the existing alternatives on several datasets for the same amount of human supervision. The source code of the project is available online[1].

1 INTRODUCTION

Denoising diffusion probabilistic models (DDPM) (Sohl-Dickstein et al., 2015; Ho et al., 2020) have recently outperformed alternative approaches to model the distribution of natural images both in the realism of individual samples and their diversity (Dhariwal & Nichol, 2021). These advantages of DDPM are successfully exploited in applications, such as colorization (Song et al., 2021), inpainting (Song et al., 2021), super-resolution (Saharia et al., 2021; Li et al., 2021b), and semantic editing (Meng et al., 2021), where DDPM often achieve more impressive results compared to GANs.

So far, however, DDPM were not exploited as a source of effective image representations for discriminative computer vision problems. While the prior literature has demonstrated that various generative paradigms, such as GANs (Donahue & Simonyan, 2019) or autoregressive models (Chen et al., 2020a), can be used to extract the representations for common vision tasks, it is not clear if DDPM can also serve as representation learners. In this paper, we provide an affirmative answer to this question in the context of semantic segmentation.

In particular, we investigate the intermediate activations from the U-Net network that approximates the Markov step of the reverse diffusion process in DDPM. Intuitively, this network learns to denoise its input, and it is not clear why the intermediate activations should capture semantic information needed for high-level vision problems. Nevertheless, we show that on the certain diffusion steps, these activations do capture such information, therefore, can potentially be used as image representations for downstream tasks. Given these observations, we propose a simple semantic segmentation method, which exploits these representations and successfully works even if only a few labeled images are provided. On several datasets, we show that our DDPM-based segmentation method outperforms the existing baselines for the same amount of supervision.

To sum up, the contributions of our paper are:

1. We investigate the representations learned by the state-of-the-art DDPM and show that they capture high-level semantic information valuable for downstream vision tasks.

[1] anonymous URL

2. We design a simple segmentation approach that exploits these representations and outperforms the alternatives in the few-shot operating point.
3. We compare the DDPM-based representations with their GAN-based counterparts on the same datasets and demonstrate the advantages of the first ones in the context of semantic segmentation.

2 RELATED WORK

In this section, we briefly describe the existing research lines relevant to our work.

Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020) are a class of generative models that approximate the distribution of real images by the endpoint of the Markov chain that originates from a simple parametric distribution, typically a standard Gaussian. Each Markov step is modeled by a deep neural network that effectively learns to invert the diffusion process with a known Gaussian kernel. (Ho et al., 2020) highlighted the equivalence of diffusion models and score matching (Song & Ermon, 2019; 2020), showing them to be two different perspectives on the gradual conversion of a simple known distribution into a target distribution via the iterative denoising process. Very recent works (Nichol & Dhariwal, 2021; Dhariwal & Nichol, 2021) have developed more powerful model architectures as well as different advance objectives for DDPM, which led to the "victory" of DDPM over GANs in terms of generative quality. DDPM have been widely used in several applications, including image colorization (Song et al., 2021), super-resolution (Saharia et al., 2021; Li et al., 2021b), inpainting (Song et al., 2021), semantic editing (Meng et al., 2021). In our work, we demonstrate that one can also successfully use them for semantic segmentation.

Image segmentation with generative models is an active research direction at the moment, however, existing methods are primarily based on GANs. The first line of works (Voynov & Babenko, 2020; Voynov et al., 2021; Melas-Kyriazi et al., 2021) is based on the evidence that the latent spaces of the state-of-the-art GANs have directions corresponding to effects that influence the foreground/background pixels differently, which allows producing synthetic data to train segmentation models. However, the methods from (Voynov & Babenko, 2020; Voynov et al., 2021; Melas-Kyriazi et al., 2021) are currently able to perform binary segmentation only, and it is not clear if they can be used in the general setup of semantic segmentation. The second line of works (Zhang et al., 2021; Tritrong et al., 2021; Xu & Zheng, 2021) is more relevant to our study since they are based on the intermediate representations obtained in GANs. In particular, the method proposed in (Zhang et al., 2021) trains a pixel class prediction model on these representations and confirms their label efficiency. In the experimental section, we compare the method from (Zhang et al., 2021) to our DDPM-based one and demonstrate several distinctive advantages of our solution.

Representations from generative models for discriminative tasks. The usage of generative models as representation learners has been widely investigated for global prediction (Donahue & Simonyan, 2019; Chen et al., 2020a), and dense prediction problems (Zhang et al., 2021; Tritrong et al., 2021; Xu & Zheng, 2021; Xu et al., 2021). While previous works highlighted the practical advantages of these representations, such as out-of-distribution robustness (Li et al., 2021a), generative models as representation learners receive less attention compared to alternative unsupervised methods, e.g., based on contrastive learning (Chen et al., 2020b). The main reason is probably the difficulty of training a high-quality generative model on a complex, diverse dataset. However, given the recent success of DDPM on Imagenet (Deng et al., 2009), one can expect that this direction will attract more attention in the future.

3 REPRESENTATIONS FROM DIFFUSION MODELS

Notation. In the following sections we investigate the pretrained DDPM, which start sampling with noise x_T and gradually produce less noisy $x_{T-1}, \dots$ until reaching the final sample x_0. Formally, we are given a forward (diffusion) process

$$q(x_t|x_{t-1}) \coloneqq \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I), \tag{1}$$

for some fixed variance schedule $\beta_1, \dots, \beta_t$. Importantly, a noisy sample x_t can be obtained directly from the clean sample x_0 as $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$, with $\epsilon \sim \mathcal{N}(0,1)$ and $\alpha_t \coloneqq 1 - \beta_t$, $\bar{\alpha}_t \coloneqq \prod_{s=1}^{t} \alpha_s$.

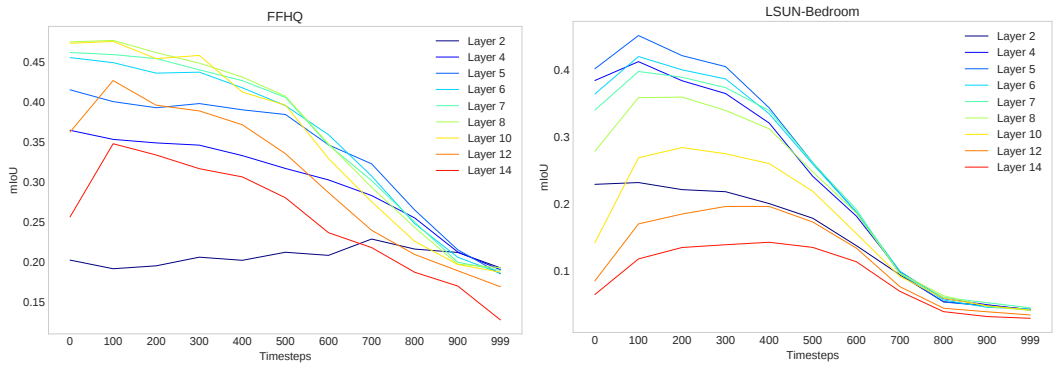


Figure 1: The evolution of predictive performance of DDPM-based pixel-wise representations for different layers and diffusion steps. The most informative features typically correspond to the later diffusion steps and middle layers of the UNet decoder. The earlier diffusion steps correspond to uninformative representations.

Pretrained DDPM provides an approximate reverse process:

$$p_\theta(x_{t-1}|x_t) := \mathcal{N}(x_t; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)). \tag{2}$$

In practice, rather than predicting the mean of the distribution in Equation (2), the noise predictor network $\epsilon_\theta(x_t, t)$ predicts the noise component from the sample x_t and the step t; the mean is then a linear combination of this noise component and x_t. The covariance predictor $\Sigma_\theta(x_t, t)$ can be either a fixed set of scalar covariances or learned as well (the latter was shown to improve the quality of models (Nichol & Dhariwal, 2021)).

The denoising model $\epsilon_\theta(x_t, t)$ is typically parameterized by different variants of the UNet architecture (Ronneberger et al., 2015), and in our experiments we investigate the state-of-the-art variant proposed in (Dhariwal & Nichol, 2021).

Extracting representations. For a given real image $x_0 \in \mathbb{R}^{H \times W \times 3}$, one can compute T sets of activation tensors from the noise predictor network $\epsilon_\theta(x_t, t)$. Formally for $t = T \ldots 1$ we first corrupt x_0 by adding a Gaussian noise: $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$. The noisy x_t is used as an input of $\epsilon_\theta(x_t, t)$ parameterized by the UNet model. The UNet's intermediate activations are then upsampled with a bilinear interpolation to the spatial dimensions equal $H \times W$. This allows to treat them as pixel-level representations of x_0.

3.1 Representation analysis

First, we analyze the representations produced by the noise predictor $\epsilon_\theta(x_t, t)$ for different t. Here, we consider the state-of-the-art DDPM checkpoints trained on the LSUN-Bedroom and FFHQ datasets[2].

The intermediate activations from the noise predictor capture semantic information. For this experiment, we take a few images from the LSUN-Bedroom/FFHQ datasets and manually assign each pixel to one of the $28/34$ semantic classes. The lists of classes for both datasets are provided in the appendix. Our goal is to understand whether the pixel-level representations produced by DDPM effectively capture the information about semantics. To this end, we train a multi-layer perceptron (MLP) to predict the pixel semantic label from its features produced by a particular UNet layer on a specific diffusion step t. MLP is trained on $40/20$ images and evaluated on $20/20$ hold-out ones. The predictive performance is measured in terms of mean IoU. The evolution of predictive performance across the different UNet layers and different diffusion steps t is presented in Figure 1. Figure 1 shows that the discriminability of the features produced by the noise predictor $\epsilon_\theta(x_t, t)$ varies for different layers and diffusion steps. In particular, the features corresponding to the later diffusion steps (the smaller values of t) typically capture semantic information more effectively, while the ones corresponding to the earlier steps are generally uninformative. Across different layers, the

[2] https://github.com/openai/guided-diffusion

Figure 2: Examples of k-means clusters (k=5) formed by the features extracted from the layers $\{6, 8, 10, 12\}$ on the diffusion steps $\{50, 250, 450, 650, 850\}$. The clusters from earlier layers and diffusion steps spatially span coherent semantic objects and parts.

features produced by the layers in the middle of the UNet decoder appear to be the most informative on all diffusion steps.

Figure 1 implies that for certain layers and steps, similar DDPM-based representations correspond to the pixels of the same semantics. Figure 2 shows the k-means clusters (k=5) formed by the features extracted from the layers $\{6, 8, 10, 12\}$ on the diffusion steps $\{50, 250, 450, 650, 850\}$, and confirms that clusters can span coherent semantic objects and object-parts. Note that on the earlier layer L=6, the features correspond to coarse semantic masks, e.g., they cannot discriminate between various face parts. On the other extreme, the features from L=12 do not exhibit semantic meaningness, reaffirming the predictive performance behavior from Figure 1. The layer L=8 appears to be a "sweet spot" demonstrating fine-grained semantic fragmentation. Across different diffusion steps, the most meaningful features correspond to the later ones. We attribute this behavior to the fact that on the earlier diffusion steps, the global structure of a DDPM sample has not emerged yet, therefore, it is hardly possible to predict segmentation masks at this stage. This intuition is qualitatively confirmed by the masks in Figure 2. For t=850, the masks poorly reflect the content of actual images, while for smaller values of t, the masks and images are semantically coherent.

3.2 DDPM-BASED REPRESENTATIONS FOR FEW-SHOT SEMANTIC SEGMENTATION

The potential effectiveness of the intermediate DDPM activations observed above implies their usage as image representations for the dense prediction task. Figure 3 schematically presents a simple segmentation approach, which exploits the discriminability of these representations. In more detail, we consider a few-shot semi-supervised setup, when a large number of unlabeled images $\{X_1, \dots, X_N\} \subset \mathbb{R}^{H \times W \times 3}$ from the particular domain are available, and only for n training images $\{X_1, \dots, X_n\} \subset \mathbb{R}^{H \times W \times 3}$ the groundtruth K-class semantic masks $\{Y_1, \dots, Y_n\} \subset \mathbb{R}^{H \times W \times \{1, \dots, K\}}$ are provided. As a first step, we train a diffusion model on the whole $\{X_1, \dots, X_N\}$ in an unsupervised manner. Then, this diffusion model is used to extract the pixel-level representations of labeled images using the subset of the UNet's layers and diffusion steps t. In this work, we use the representations from the layers L=$\{6, 7, 8\}$ and steps t=$\{100, 200, 300, 400\}$. The extracted representations from all layers and steps t are upsampled to the image size and concatenated, forming the feature vectors for all pixels of training images. Then, following Zhang et al. (2021), we train an ensemble of 10 independent multi-layer perceptrons (MLPs) on these feature vectors, which aim to predict a semantic label of each pixel available

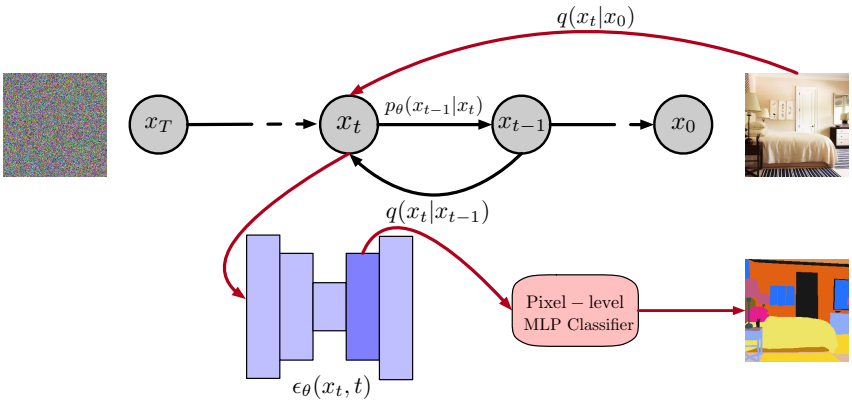


Figure 3: Extracting the pixel-level image representations from the pretrained DDPM. The intermediate activations from the noise predictor UNet are used to train the pixel-level segmentation model parameterized by MLP. When segmenting a test image, its DDPM-based representations are used to predict the pixel class labels.

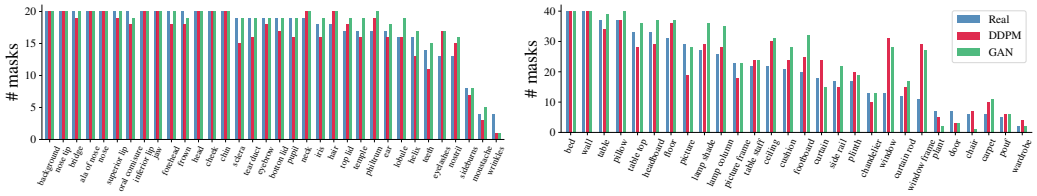
for training images. The exact architecture of MLPs used in our experiments is reported in the appendix. To segment a test image, its DDPM-based pixel-wise representations are extracted from DDPM, and these representations are used to predict the pixel labels by MLPs, the final prediction is obtained by majority voting.

4 Experiments

This section experimentally confirms the advantage of the DDPM-based representations for the semantic segmentation problem. We start from a thorough comparison to the existing alternatives and then dissect the reasons for the DDPM success by additional analysis.

Methods. In the evaluation, we compare our method to several prior approaches, which tackle the few-shot semantic segmentation setup.

- **DatasetGAN** (Zhang et al., 2021) — this method is the relevant baseline to ours since it exploits the discriminability of pixel-level features produced by GANs. In more detail, this method employs human assessors to provide segmentation masks for a few GAN-produced images. The latent codes of these images are then used to obtain the intermediate generator activations, which are considered pixel-level representations. Given these representations, a classifier is trained to predict a semantic label for each pixel. This classifier is then used to label new synthetic GAN images, which, for their part, serve as a training set for the DeepLabV3 segmentation model (Chen et al., 2017). In our experiments, we use the official authors' implementation and protocol from (Zhang et al., 2021).
- **DatasetDDPM** — this method mirrors the previous **DatasetGAN** baseline with the only difference that GANs are replaced with DDPM models. We include this baseline to compare the GAN-based and DDPM-based representations in the same scenario. Note that our segmentation method described in Section 3 is more straightforward compared to **DatasetGAN** and **DatasetDDPM** since it does not require auxiliary steps of the synthetic dataset generation and training the segmentation model on it.
- **GAN Inversion** — this baseline employs the state-of-the-art GAN inversion method (Tov et al., 2021) to obtain the latent codes for real images. In more detail, we map the annotated real images to the GAN latent space, which allows computing the intermediate generator activations and using them as pixel-level representations. These representations are then used to train the pixel classifier. We include this baseline since it does not require the usage of synthetic data, which can potentially be beneficial due to the absence of domain gap.
- **Supervised Pretrain** — in this baseline, we finetune the last layer of the pretrained DeepLabV3 network on the labeled images from our datasets. The initial weights of



(a) FFHQ (b) LSUN-Bedrooms

Figure 4: Number of masks for each semantic class across real and synthetic images.

DeepLabV3 are pretrained on the semantic segmentation task on MS-COCO (Lin et al., 2014). This method is the same as the "Transfer Learning" baseline in (Zhang et al., 2021).

- **Self-Supervised Pretrain** — this baseline uses the representations produced by the recent self-supervised learning methods (He et al., 2020; Caron et al., 2020). Here we consider the MOCO model (He et al., 2020) trained on the face images and the SwAV model (Caron et al., 2020) pretrained on Imagenet for the LSUN datasets. The upsampled intermediate activations from every second bottleneck block of the backbone are used as pixel-level representations. Then we train the pixel classifier on these representations in the same way as in the baselines above.
- **DDPM** — the proposed method that was described in Section 3.

Datasets. In our evaluation, we work with the LSUN-Bedroom, FFHQ-256, and LSUN-Cat datasets. As a training set for each dataset, we consider several images for which the fine-grained semantic masks annotations are collected following the protocol from (Zhang et al., 2021). Specifically, for each dataset, the professional assessor was hired to annotate train and test samples. The semantic classes used to segment the images from each domain are listed in the appendix. The methods use the same number of annotated images for training and the same set of images for evaluation. We report the number of annotated images used for each dataset in Table 1. The pixel-classifiers for **DatasetGAN** and **DatasetDDPM** baselines are trained on the annotated synthetic images produced by GAN and DDPM, respectively. All other methods are trained on real images. Evaluation of all methods is performed on real images.

In Figure 4, we report the statistics of pixel classes computed over annotated real images as well as annotated synthetic images produced by GAN and DDPM. Figure 4 shows that GAN-based and DDPM-based synthetics have comparable coverage of most semantic classes.

Dataset	Classes	$\text{Real}_{\text{Train}}$	$\text{Real}_{\text{Test}}$	GAN	DDPM	Total
Bedroom	28	40	20	40	40	140
FFHQ	34	20	20	20	20	80
LSUN-Cat	16	30	20	30	30	110

Table 1: The number of annotated images used in our experiments. For each dataset, we annotate the real images for training and evaluation, and also annotate GAN-produced and DDPM-produced synthetic images to train the DatasetGAN and DatasetDDPM baselines respectively.

Generative Models. In our experiments, we use the state-of-the-art StyleGAN2 (Karras et al., 2020) models for the GAN-based baselines and the state-of-the-art pretrained DDPM models from (Dhariwal & Nichol, 2021) for our DDPM-based method. (Dhariwal & Nichol, 2021) does not provide a pretrained DDPM model for FFHQ, so we train it ourselves using the official implementation from (Dhariwal & Nichol, 2021)[3].

Main results. The comparison of the methods in terms of the mean IoU measure is presented in Table 2. We also depict per class IoUs for FFHQ and LSUN-Bedroom datasets in Figure 5. Additionally, we provide several qualitative examples of segmentation with our method in Figure 6. Below we highlight several key observations:

[3] https://github.com/openai/guided-diffusion

- The proposed method based on the DDPM representations outperforms the alternatives on all three datasets. The most significant advantage is achieved on the more complex LSUN datasets, while on the simpler FFHQ dataset, the margin is less notable.
- **DatasetDDPM** outperforms its counterpart **DatasetGAN** on all benchmarks. Note that both these methods use the DeepLabV3 network, trained on the same amount of synthetics produced by the corresponding generative model. We attribute this superiority to the higher quality of DDPM synthetics, therefore, the smaller domain gap between synthetic and real data.
- On two LSUN datasets, **DDPM** outperforms the **DatasetDDPM** competitor, while on the simpler FFHQ, their performance is almost the same. We provide explanations and additional experiments on this in the analysis section below.
- The **Self-Supervised Pretrain** baseline underperforms compared to the DDPM-based segmentation. We attribute this behavior to the fact that this baseline is trained in the discriminative fashion and can suppress the details, which are needed for fine-grained semantic segmentation. This result is consistent with the recent findings in (Cole et al., 2021), which shows that the state-of-the-art contrastive methods produce representations, which are suboptimal for fine-grained problems.
- The **GAN Inversion** method performs poorly on the LSUN datasets. For these relatively complex domains, the GAN inverter cannot invert most real images reliably. On the simpler FFHQ dataset, **GAN Inversion** is competitive and even outperforms the **DatasetGAN** baseline. This result highlights the importance of using real images in the setups when high-quality generative models are available.

Overall, the proposed DDPM-based segmentation outperforms the baselines that exploit alternative generative models and also the baselines trained in the discriminative fashion. This result highlights the potential of using the state-of-the-art DDPM as strong unsupervised representation learners.

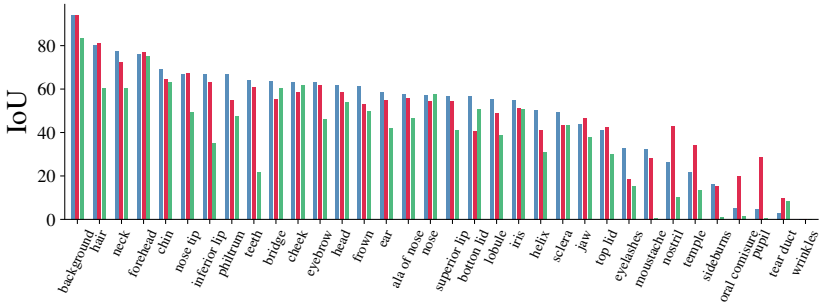
Method	LSUN-Bedroom	FFHQ-256	LSUN-Cat
Supervised Pretrain	19.4	23.5	20.9
Self-Supervised Pretrain	30.2	42.9	17.1
DatasetGAN	29.1	37.8	21.1
DatasetDDPM	33.9	**49.8**	31.4
GAN Inversion	13.9	41.2	11.1
DDPM	**45.8**	**48.7**	**48.5**

Table 2: The comparison of the segmentation methods in terms of mean IoU on the LSUN-Bedroom, FFHQ-256 and LSUN-Cat datasets.

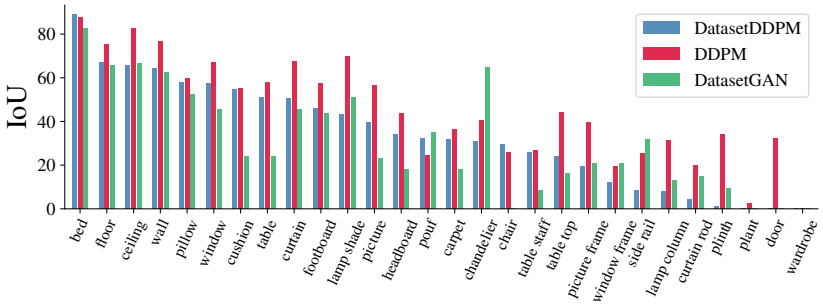
4.1 Analysis

DDPM vs DatasetDDPM. Both these methods exploit the DDPM-based representations, however, the performance of DatasetDDPM is significantly lower on the LSUN datasets. We explain this behavior by several factors:

- First, DDPM is trained on the annotated real images, while DatasetDDPM is trained on the annotated synthetic images, which are typically less natural, diverse, and can lack objects of particular classes. The contribution from this factor is quantified by Table 3, which compares two methods both trained either on real images or on synthetic images. As can be seen, training on real images improves the DatasetDDPM performance on LSUN-Cat but not on other datasets.
- The second factor originates from the potential train/evaluation domain gap since DatasetDDPM is trained on synthetic images and evaluated on real images. The contribution from this factor is revealed in Table 4, which shows that when the methods are evaluated on DDPM samples, the performance margin becomes smaller.



(a) FFHQ



(b) LSUN-Bedrooms

Figure 5: Per class IoUs for datasetGAN, datasetDDPM and DDPM

	LSUN-Bedroom		FFHQ-256		LSUN-Cat	
Train on	Real	DDPM samples	Real	DDPM samples	Real	DDPM samples
DatasetDDPM	34.8	34.4	46.9	49.8	35.5	31.4
DDPM	45.0	45.1	48.7	50.9	48.5	36.4

Table 3: Performance of DDPM and DatasetDDPM segmentation when trained on real and synthetic images. Even when trained on synthetic images, DDPM achieves higher performance.

- Finally, we attribute the higher performance of DDPM to the fact that the UNet models used to extract the representations are more powerful compared to DeeplabV3, which is used in DatasetDDPM. From a practical standpoint, this can be a limitation of DDPM segmentation since it requires much more computation during inference. The possibility to "distill" its effectiveness into more lightweight architectures is an interesting question for future research.

Method	LSUN-Bedroom		FFHQ-256		LSUN-Cat	
Evaluate on	Real	DDPM samples	Real	DDPM samples	Real	DDPM samples
DatasetDDPM	34.8	40.4	46.9	46.8	35.5	42.2
DDPM	45.0	49.6	48.7	49.3	48.5	46.5

Table 4: Performance of DDPM and DatasetDDPM segmentation when evaluated on real and synthetic images. DDPM achieves higher performance on both real and synthetic test samples.

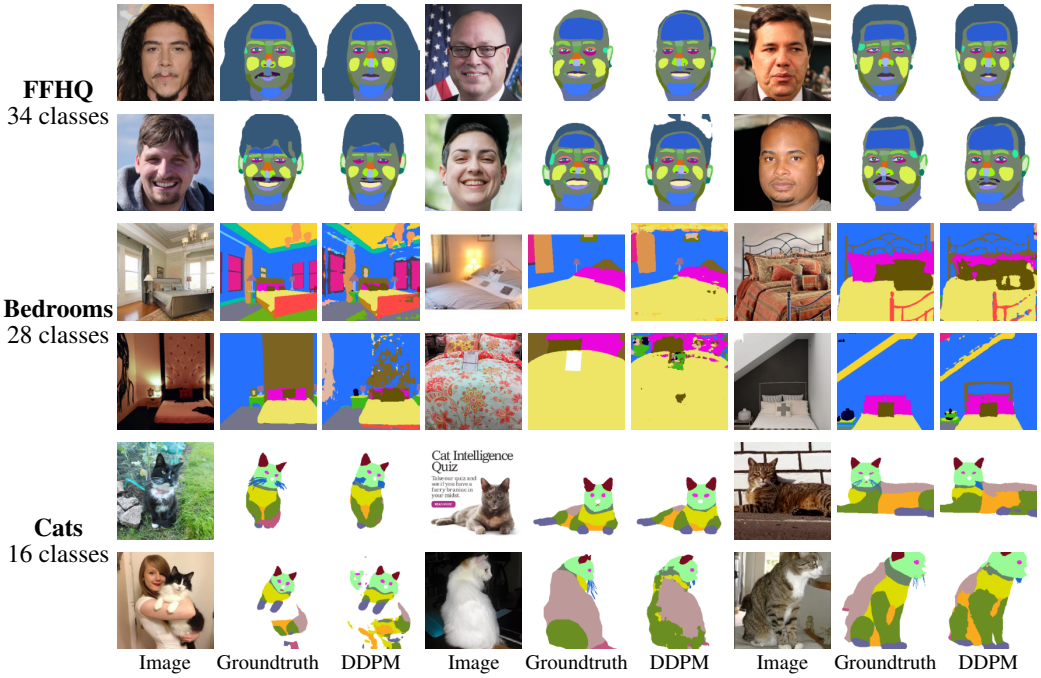


Figure 6: The examples of segmentation masks predicted by our method on the FFHQ, LSUN-Bedrooms and LSUN-Cats test images along with the groundtruth annotated masks.

Sample-efficiency. In this experiment, we evaluate the performance of our method when it utilizes less amount of annotated data. We provide mIoU for three dataset in Table 5. Importantly, DDPM is able to outperform the DatasetGAN and DatasetDDPM approaches using only half of the data.

LSUN-Bedroom				FFHQ				LSUN-Cat			
40	30	20	10	20	16	12	8	30	24	16	8
0.451	0.431	0.361	0.297	0.509	0.508	0.499	0.485	0.485	0.447	0.354	0.325

Table 5: Evaluation of the proposed method with different number of labeled training data.

5 Conclusion

This paper demonstrates that DDPM can serve as representation learners for discriminative computer vision problems. Compared to GANs, diffusion models allow for a straightforward computation of these representations for real images, and one does not need to learn an additional encoder, which maps images to the latent space. This DDPM's advantage and superior generative performance provide state-of-the-art performance in the few-shot semantic segmentation task. The notable restraint of the DDPM-based segmentation is a requirement of high-quality diffusion models trained of the dataset at hand, which can be challenging for complex domains, like Imagenet or MSCOCO. However, given the rapid research progress on DDPM, we expect they will reach these milestones in the nearest future, thereby extending the range of applicability for the corresponding representations.

REFERENCES

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.

Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *ICML*, 2020a.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020b.

Elijah Cole, Xuan Yang, Kimberly Wilber, Oisin Mac Aodha, and Serge Belongie. When does contrastive visual representation learning work? *arXiv preprint arXiv:2105.05837*, 2021.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. 2021.

Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. *NeurIPS*, 2019.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. 2020.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8107–8116, 2020.

Daiqing Li, Junlin Yang, Karsten Kreis, Antonio Torralba, and Sanja Fidler. Semantic segmentation with generative models: Semi-supervised learning and strong out-of-domain generalization. In *CVPR*, 2021a.

Haoying Li, Yifan Yang, Meng Chang, Huajun Feng, Zhihai Xu, Qi Li, and Yueting Chen. Srdiff: Single image super-resolution with diffusion probabilistic models. 2021b.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Finding an unsupervised image segmenter in each of your deep generative models. *arXiv preprint arXiv:2105.08127*, 2021.

Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. 2021.

Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. *ICML*, 2021.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. 2021.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *NeurIPS*, 2019.

Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *NeurIPS*, 2020.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. 2021.

Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *arXiv preprint arXiv:2102.02766*, 2021.

Nontawat Tritrong, Pitchaporn Rewatbowornwong, and Supasorn Suwajanakorn. Repurposing gans for one-shot semantic part segmentation. In *CVPR*, 2021.

Andrey Voynov and Artem Babenko. Unsupervised discovery of interpretable directions in the gan latent space. In *ICML*, 2020.

Andrey Voynov, Stanislav Morozov, and Artem Babenko. Object segmentation without labels with large-scale generative models. *ICML*, 2021.

Jianjin Xu and Changxi Zheng. Linear semantics in generative adversarial networks. In *CVPR*, 2021.

Yinghao Xu, Yujun Shen, Jiapeng Zhu, Ceyuan Yang, and Bolei Zhou. Generative hierarchical features from synthesizing images. In *CVPR*, 2021.

Yuxuan Zhang, Huan Ling, Jun Gao, Kangxue Yin, Jean-Francois Lafleche, Adela Barriuso, Antonio Torralba, and Sanja Fidler. Datasetgan: Efficient labeled data factory with minimal human effort. In *CVPR*, 2021.

6 Appendix

6.1 List of classes

We list all the classes used for semantic segmentation:

FFHQ: [background, head, cheek, chin, ear, helix, lobule, botton lid, eyelashes, iris, pupil, sclera, tear duct, top lid, eyebrow, forehead, frown, hair, sideburns, jaw, moustache, inferior lip, oral comisure, superior lip, teeth, neck, nose, ala of nose, bridge, nose tip, nostril, philtrum, temple, wrinkles]

LSUN-Bedrooms: [bed, footboard, headboard, side rail, carpet, ceiling, chandelier, curtain, cushion, floor, table, table top, picture, pillow, lamp column, lamp shade, wall, window, curtain rod, window frame, chair, picture frame, plinth, door, pouf, wardrobe, plant, table staff]

LSUN-Cats: [background, cat, back, belly, chest, leg, paw, head, ear, eye, mouth, tongue, nose, tail, whiskers, neck]

6.2 MLP architecture

On the FFHQ dataset we use an MLP with two hidden layers of size 256 and 128 with ReLU nonlinearity and batch normalization.

On the LSUN datasets we use an MLP with two hidden layers of size 128 and 32 with ReLU nonlinearity and batch normalization.