Denoising Pretraining for Semantic Segmentation

Emmanuel Asiedu Brempong[†], Simon Kornblith, Ting Chen
Niki Parmar, Matthias Minderer[*], Mohammad Norouzi[*]
Google Research
{brempong, skornblith, iamtingchen, nikip, mjlm, mnorouzi}@google.com

Abstract

Semantic segmentation labels are expensive and time consuming to acquire. To improve label efficiency of semantic segmentation models, we revisit denoising autoencoders and study the use of a denoising objective for pretraining UNets. We pretrain a Transformer-based UNet as a denoising autoencoder, followed by fine-tuning on semantic segmentation using few labeled examples. Denoising pretraining outperforms training from random initialization, and even supervised ImageNet-21K pretraining of the encoder when the number of labeled images is small. A key advantage of denoising pretraining over supervised pretraining of the backbone is the ability to pretrain the decoder, which would otherwise be randomly initialized. We thus propose a novel Decoder Denoising Pretraining (DDeP) method, in which we initialize the encoder using supervised learning and pretrain only the decoder using the denoising objective. Despite its simplicity, DDeP achieves state-of-the-art results on label-efficient semantic segmentation, offering considerable gains on the Cityscapes, Pascal Context, and ADE20K datasets.

1. Introduction

Many important problems in computer vision, such as semantic segmentation and depth estimation, entail dense pixel-level predictions. Building accurate supervised models for these tasks is challenging because collecting ground truth annotations densely across all image pixels is costly, time-consuming, and error-prone. Hence, state-of-the-art techniques often resort to pretraining, where the model backbone (*i.e.,* encoder) is first trained as a supervised classifier [51, 68, 79] or a self-supervised feature extractor [4, 16, 36, 37, 40, 64]. Most backbone architectures, such as ResNets [38], gradually reduce the feature map resolution. Hence, to conduct pixel-level prediction, one introduces a decoder with a number of upsampling layers and additional parameters. Most state-of-the-art semantic segmentation models ignore decoder parameters and initialize them at random.

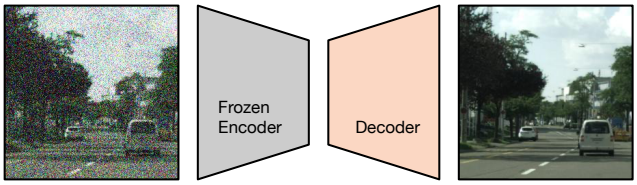


Figure 1. An illustration of decoder denoising pretraining (DDeP). Similar to a standard denoising autoencoder, the network is trained to denoise a noisy input image. However, the encoder is pretrained using supervised learning and is kept frozen, and only the parameters of the decoder are optimized using the denoising objective. Furthermore, given a noisy input denoted $\boldsymbol{x} + \sigma\epsilon$, the decoder is trained to predict noise ϵ instead of the clean image $\boldsymbol{x}$ directly.

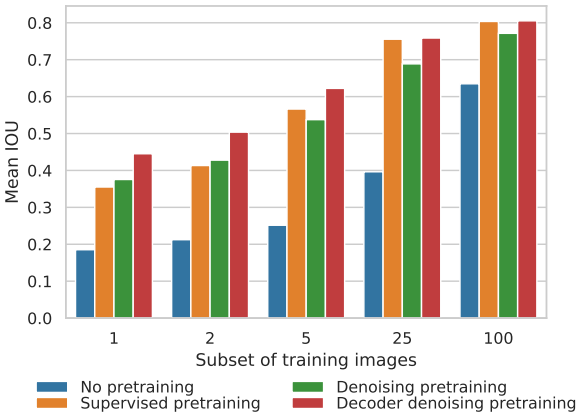


Figure 2. Mean IoU on the Cityscapes validation set as a function of fraction of labeled training images available. Denoising pretraining is particularly effective when less than 5% of labeled images is available. Supervised pretraining of the backbone on ImageNet-21K outperforms denoising pretraining when label fraction is larger. Decoder denoising pretraining offers the best of both worlds, achieving competitive results across label fractions.

An alternative avenue for pretraining semantic segmen-

[†]Work done as part of the Google AI Residency.
[*]Equal advising.

tation architectures is generative modeling, in which large amounts of unlabeled data are used to learn the data distribution. Generative modeling is promising as a representation learning method for dense prediction tasks since it does not require labels and typically learns to represent images at the pixel-level. Generative pretraining has been very successful for language tasks, where models are pretrained via conditional generation of masked tokens [19, 54, 56, 69]. However, generative pretraining for computer vision [5, 14, 24, 77] still tends to fall behind supervised pretraining, especially in the context of dense pixel prediction tasks [107].

Recently, diffusion and score-based generative models [41, 80, 81] have emerged as a new approach to image and audio synthesis [15, 22, 42, 61, 76], outperforming strong GAN and autoregressive baselines in sample quality scores. Denoising Diffusion Probabilistic Models (DDPMs) [41] approximate complex empirical distributions by learning to convert Gaussian noise to the target distribution via a sequence of iterative denoising steps. In practice, DDPMs are implemented as encoder-decoder architectures (*e.g.,* UNets [73]) that are trained to iteratively recover successively cleaner images from noise-corrupted inputs. DDPMs are therefore architecturally similar to dense prediction models, making their representations good candidates for tasks like semantic segmentation.

Inspired by the success of diffusion models, we investigate the effectiveness of representations learned by denoising autoencoders for semantic segmentation. We find that Denoising Pretraining (DeP) with a fixed, carefully chosen noise level, yields representations that are very competitive on few-shot semantic segmentation, outperforming training from random initialization, and even supervised ImageNet-21K pretraining of the encoder. However, as the number of labeled examples increases, supervised pretraining of the backbone becomes more competitive than DeP, *e.g.,* see Figure 2.

We simply add scaled i.i.d. Gaussian noise to images and train the encoder-decoder architecture to recover noise. As shown in Figure 2, DeP is very competitive at low label regimes, but it underperforms supervised pretraining at higher label regimes. To tackle this, we propose decoder denoising pretraining (DDeP) that combines supervised pretraining of the backbone with denoising pretraining of the decoder and offers competitive results across various label regimes.

Our key contributions include:

- We propose unsupervised Denoising Pretraining (DeP) for semantic segmentation, which greatly outperforms training from random initialization, and even outperforms supervised ImageNet-21K pretraining when the number of labeled images is small.
- We propose decoder denoising pretraining (DDeP) as a way to combine the benefits of supervised pretraining of backbones and denoising pretraining – A frozen supervised backbone is used in conjunction with a decoder that is pretrained on denoising.

2. Related work

Because collecting detailed pixel-level labels for semantic segmentation is costly, time-consuming, and error-prone, many methods have been proposed to enable semantic segmentation from fewer labeled examples [29, 32, 44, 48, 58, 59, 63, 65, 86, 104, 108]. These methods often resort to semi-supervised learning (SSL) [9, 87], in which one assumes access to a large dataset of unlabeled images in addition to labeled training data. In what follows, we will discuss previous work on the role of strong data augmentation, generative models, self-training, and self-supervised learning in label-efficient semantic segmentation. While this work focuses on self-supervised pretraining, we believe strong data augmentation and self-training can be combined with the proposed denoising pretraining approach to improve the results even further.

Data augmentation. French *et al.* [31] demonstrate that strong data augmentation techniques such as Cutout [21] and CutMix [98] are particularly effective for semantic segmentation from few labeled examples. Ghiasi *et al.* [33] find that a simple copy-paste augmentation is helpful for instance segmentation. Previous work [3, 6, 13, 72] also explores completely unsupervised semantic segmentation by leveraging GANs [35] to compose different foreground and background regions to generate new plausible images. We make use of relatively simple data augmentation including horizontal flip and random inception-style crop [85]. Using stronger data augmentation is left to future work.

Generative models. Early work on label-efficient semantic segmentation uses GANs to generate synthetic training data [83] and to discriminate between real and predicted segmentation masks [45, 58]. DatasetGAN [101] shows that modern GAN architectures [47] are effective in generating synthetic data to help pixel-level image understanding, when only a handful of labeled images are available. Diffusion models have been used to iteratively refine semantic segmentation masks, as a proof of concept [43]. Concurrent work [2] demonstrates the effectiveness of features learned by diffusion models for semantic segmentation from few labeled examples. By contrast, we utilize simple denoising pretraining for representation learning and study full fine-tuning of the encoder-decoder architecture as opposed to extracting fixed features [2]. Further, we use well-established benchmarks to compare our results with prior work.

Self-training, consistency regularization. *Self-training* (self-learning or pseudo-labeling) is one of the oldest SSL algorithms [1, 30, 78, 97]. It works by using an initial supervised model to annotate unlabeled data with so-called

pseudo labels, and then uses a mixture of pseudo- and human-labeled data to train improved models. This iterative process may be repeated multiple times. Self-training has been used to improve object detection [74, 107] and semantic segmentation [11, 28, 106, 108]. Consistency regularization is closely related to self-training and enforces consistency of predictions across augmentations of an image [31, 49, 65]. These methods often require careful hyperparameter tuning and a reasonable initial model to avoid propagating noise. Combining self-training with denosing pretraining will likely improve the results further.

Self-supervised learning. Self-supervised learning methods formulate predictive pretext tasks that are easy to construct from unlabeled data and can benefit downstream discriminative tasks. In natural language processing (NLP), the task of masked language modeling [20, 56, 70] has become the de facto standard, showing impressive results across NLP tasks. In computer vision, different pretext tasks for self-supervised learning have been proposed, including the task of predicting the relative positions of neighboring patches within an image [23], the task of inpainting [66], solving Jigsaw Puzzles [62], image colorization [53, 99], rotation prediction [34], and other tasks [8, 52, 100]. Recently, methods based on exemplar discrimination and contrastive learning have shown promising results for image classification [16, 36, 37, 40, 64]. These approaches have been used to successfully pretrain backbones for object detection and segmentation [17, 37], but unlike this work, they often initialize decoder parameters at random.

Self-supervised learning for dense prediction. Pinheiro *et al.* [67] and Wang *et al.* [96] propose dense contrastive learning, an approach to self-supervised pretraining for dense prediction tasks, in which contrastive learning is applied to patch- and pixel-level features as opposed to image level-features. This is reminiscent of AMDIM [4] and CPC V2 [39]. Zhong *et al.* [104] take this idea further and combine segmentation mask consistency between the output of the model for different augmentations of an image (possibly unlabeled) and pixel-level feature consistency across augmentations. Perhaps, BeIT [5] is most related to this paper, which applies masked auto-encoding, the key idea of BERT, to pretraining vision transformers for image classification and segmentation. BeIT uses softmax over discretized image patches [71] to conduct patch inpainting. Our approach is simpler and does not require quantizing image patches using a separate model. Further, we show promising results on label-efficient semantic segmentation. We revisit one of the oldest and simplest self-supervised learning methods, namely denoising autoencoders [90, 92]. We show that when modern U-Net architectures are used to denoise a carefully selected level of Gaussian noise, the learned representations transfer favorably to label-efficient semantic segmentation.

Diffusion models. Diffusion and score-based generative models [41, 80, 81] represent an emerging family of generative models resulting in image sample quality superior to GANs [22, 42]. These models are linked to denoising autoencoders through denoising score matching [89] and can be seen as methods to train energy-based models [46]. Denoising Diffusion Models (DDPMs) have recently been applied to conditional generation tasks such as super-resolution, colorization, and inpainting [55, 75, 76, 82], suggesting these models may be able to learn useful image representations. We are inspired by the success of DDPMs, but we find that many components of DDPMs are not necessary and simple denoising pretraining works well.

Transformers for vision. Inspired by the success of Transformers in NLP [88], several publications study combining convolution and self-attention for object detection [7], semantic segmentation [94, 95], and panoptic segmentation [93]. Vision Transformer (ViT) [25] demonstrates that a convolution-free approach can yield impressive results when a massive labeled dataset is available. Recent work has explored the use of ViT as a backbone for semantic segmentation [57, 84, 103]. These approaches differ in the structure of the decoder, but they show the power of Vit-based semantic segmentation. We adopt a hybrid ViT [26] as the backbone, where the patch embedding projection is applied to patches extracted from a convolutional feature map. We study the size of the decoder, and find that wider decoders often improve semantic segmentation results.

3. Denoising pretraining

Our goal is to learn self-supervised image representations that transfer well to dense prediction tasks such as semantic segmentation. Inspired by the recent success of denoising diffusion probabilistic models in image generation [41, 61], we revisit denoising objectives for unsupervised representation learning and adapt them to modern semantic segmentation architectures based on Transformers.

Network architecture. We use a Transformer-based UNet architecture (*a.k.a* TransUNet) [10], which integrates a 12-layer Transformer into a standard UNet [73] model. As depicted in Figure 3, the encoder is a hybrid model comprising convolution and self-attention layers [26], where patch embeddings are extracted from a CNN feature map. We adopt an encoder identical to the Hybrid-ViT model of Dosovitskiy *et al.* [26], which enables us to take advantage of supervised model checkpoints pretrained on the ImageNet-21K dataset.

In our experiments, the CNN feature extractor is a ResNet-50 model. The decoder is a standard UNet decoder with two 3x3 convolutions in each decoder block. Skip connections from feature maps in the encoder are added to the decoder to enable precise localization. We report results for

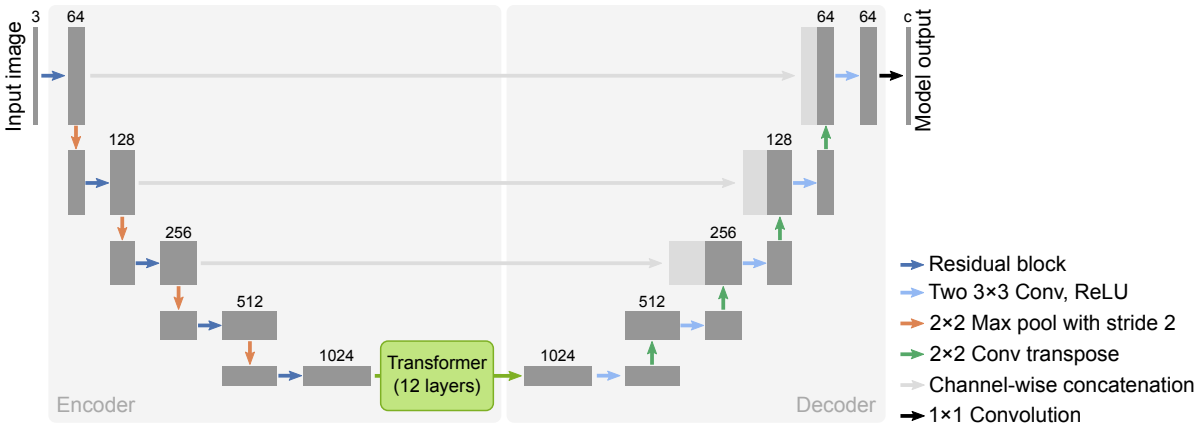


Figure 3. The Transformer-based UNet architecture used in our experiments. The encoder is a Hybrid-ViT model [26].

the basic decoder in Figure 3 and decoders with 2× and 3× as many channels. Our denoising pretraining approach is not specific to any choice of model architecture, but all of our results are reported on the TransUNet architecture.

3.1. Denoising objective function

We aim to pretrain an encoder-decoder architecture, denoted f, which is parameterized by θ. This model takes as input an image $\boldsymbol{x} \in \mathbb{R}^{H \times W \times C}$ and converts it into a dense representation $\boldsymbol{y} \in \mathbb{R}^{h \times w \times c}$, *e.g.,* a semantic segmentation mask. We wish to find an effective way to initialize the parameters θ such that f can be effectively fine-tuned on semantic segmentation on a few labeled examples.

We revisit the old and simple idea of pretraining an encoder-decoder architecture as a denoising autoencoder [91, 92] – given an unlabeled input image $\boldsymbol{x}$, the model is trained to reconstruct $\boldsymbol{x}$ from a noise-corrupted version $\boldsymbol{x} + \boldsymbol{\epsilon}$. In early work on denoising autoencoders, the decoder is removed after representation learning and only the encoder is fine-tuned on classification tasks [91]. For dense prediction tasks, however, both the encoder and the decoder can be pretrained and fine-tuned jointly.

Recently, denoising autoencoders have received renewed attention in the form of Denoising Diffusion Probabilistic Models (DDPMs; [41]). The key difference between DDPMs and denoising autoencoders is that DDPMs are trained to remove Gaussian noise added to images, where the noise is drawn from a Gaussian with varying variances. In contrast, denosing autoencoders are typically trained to remove Gaussian noise with a fixed variance. Perhaps less importantly, DDPM architectures are often conditioned on the noise level too. Further, DDPMs train an autoencoder to predict noise instead of the clean image directly.

Given the resurgence of denoising autoencoders, this paper investigates the effectiveness of representations learned by denoising autoencoders for semantic segmentation. One appealing property of denoising autoencoders is that they can make use of unlabeled data. Given an unlabeled image $\boldsymbol{x}$, we obtain a noisy image $\widetilde{\boldsymbol{x}}$ by adding Gaussian noise $\boldsymbol{\epsilon}$

$$\widetilde{\boldsymbol{x}} = \boldsymbol{x} + \sigma\boldsymbol{\epsilon}\,, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})\,. \tag{1}$$

Hence, we study the use of the DDPM denosing objective for pretraining semantic segmentation models. Given an unlabeled input image $\boldsymbol{x}$ and a scalar noise level γ, DDPMs generate a noisy image $\widetilde{\boldsymbol{x}}$ as

$$\widetilde{\boldsymbol{x}} = \sqrt{\gamma}\,\boldsymbol{x} + \sqrt{1-\gamma}\,\boldsymbol{\epsilon}\,, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})\,. \tag{2}$$

$\boldsymbol{x}$ is attenuated by $\sqrt{\gamma}$ and $\boldsymbol{\epsilon}$ is attenuated by $\sqrt{1-\gamma}$ to ensure that the variance of the random variables $\widetilde{\boldsymbol{x}}$ is 1 if the variance of $\boldsymbol{x}$ is 1. However for representation learning, we found that this formulation doesn't provide significant benefits over the denoising autoencoder formulation in Eq. (1) which is used for the rest of this paper.

Our denoising pretraining objective can be expressed as

$$\mathbb{E}_{\boldsymbol{x}}\, \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\boldsymbol{I})}\, \mathbb{E}_{\sigma\sim p(\sigma)} \Big\| f_\theta(\boldsymbol{x} + \sigma\boldsymbol{\epsilon}) - \boldsymbol{\epsilon} \Big\|_2^2\,, \tag{3}$$

where f_θ represents an image-to-image translation architecture such as a U-Net and $p(\sigma)$ defines the noise schedule for a DDPM.

In practice, we find that high-quality representations can be learned with a simple denoising objective in which σ is fixed to a single value. Hence, we drop the expectation over σ, resulting in a simpler denosing objective function:

$$\mathbb{E}_{\boldsymbol{x}}\, \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\boldsymbol{I})} \Big\| f_\theta(\boldsymbol{x} + \sigma\boldsymbol{\epsilon}) - \boldsymbol{\epsilon} \Big\|_2^2\,. \tag{4}$$

This objective can be thought of as a single iteration of the diffusion process modeled by DDPMs. We use this objective function to pretrain the whole encoder-decoder architecture after random initialization, which we dub Denoising Pretraining (DeP). Alternatively, to combine the advantages of supervised and denoising pretraining, in Decoder

Denoising Pretraining (DDeP), we initialize the backbone with classification-pretrained weights and pretrain only the decoder weights with denoising.

Inspired by [41, 89], we define the denoising objective in Eq. (4) in terms of predicting the noise vector $\boldsymbol{\epsilon}$. Alternatively, one could regress the output of f_θ onto the noiseless image $\boldsymbol{x}$ itself. The difference between these two formulations can be linked to the addition of a skip connection from the input $\widetilde{\boldsymbol{x}}$ to the output, so that the model can easily combine its estimate of $\boldsymbol{\epsilon}$ with the input $\widetilde{\boldsymbol{x}}$ to obtain $\boldsymbol{x}$. In this case both formulations should behave similarly. In the absence of an explicit skip connection, our experiments suggest that predicting $\boldsymbol{\epsilon}$ works better than predicting $\boldsymbol{x}$ (Table 7).

We find that the choice of σ has a big impact on the quality of representations. For visual inspection, Figure 4 illustrates a few reasonable values of σ for denoising pretraining. Following [41] we map pixel values to the range $[-1, 1]$ so pixels approximately conform to a zero mean and unit variance. After pretraining, we discard the final projection layer before finetuning on semantic segmentation.

3.2. Extension to diffusion process

In its simplest form, when using a single fixed value of σ in Eq. (4), our method corresponds to a single step in a diffusion process. We also study extensions that bring the method closer to the full diffusion process used in DDPMs.

Variable noise schedule. In DDPMs, which model a complete diffusion process from a clean image to pure noise (and its reverse), σ is sampled uniformly at random from $[0, 1]$ for each training example. While we find that a fixed σ often performs best, we also experiment with sampling σ randomly. In this case, restricting σ to a range close to 1 is essential for representation quality (see Section 4.5).

Conditioning on noise level. In the diffusion formalism, the model represents the (reverse) transition function from one noise level to the next, and is therefore conditioned on the current noise level. In practice, this is achieved by supplying the σ sampled for each training example as an additional model input, e.g. to normalization layers. Since we typically use a fixed noise level, conditioning is not required for our method.

Weighting of noise levels. In DDPMs, the relative weighting of different noise levels in the loss has a large impact on sample quality [41]. Since our experiments suggest that multiple noise levels are not necessary for learning transferable representations, we did not experiment with the weighting of different noise levels, but note that this may be an interesting direction for future research.

3.3. Hyperparameters and design choices

Noise magnitude. The only hyperparameter specific to denosing pretraining is the choice of noise level σ. We find that smaller σ values (*i.e.,* close to clean images) work best for Decoder Denoising Pretraining whiles higher values of σ work best for Denoising Pretraining . We report results for a sweep over different σ values in Section 4.5.2, but generally, we find that $\sigma = 0.8$ and $\sigma = 0.1$ works well across different settings and datasets for DeP and DDeP respectively.

4. Experimental results

We assess the effectiveness of the proposed Denoising Pretraining (**DeP**) and Decoder Denoising Pretraining (**DDeP**) on several semantic segmentation datasets and conduct label-efficiency experiments.

4.1. Datasets and metrics

Cityscapes [18]. This dataset contains 5000 images with high quality pixel-level annotations. Following what has become the standard practice [12, 18, 102], 19 labels are used for training and evaluation while the void label is ignored.

Pascal Context [60] is a challenging dataset with 4,998 images for training and 5,105 images for testing. There are 59 semantic classes and 1 background class.

ADE20K [105] is a scene parsing dataset, with fine-grained labels covering 150 objects and stuff categories. It has 20,210 training and 2,000 validation images.

Metrics. We report the mean Intersection of Union (mIoU) averaged over all semantic categories.

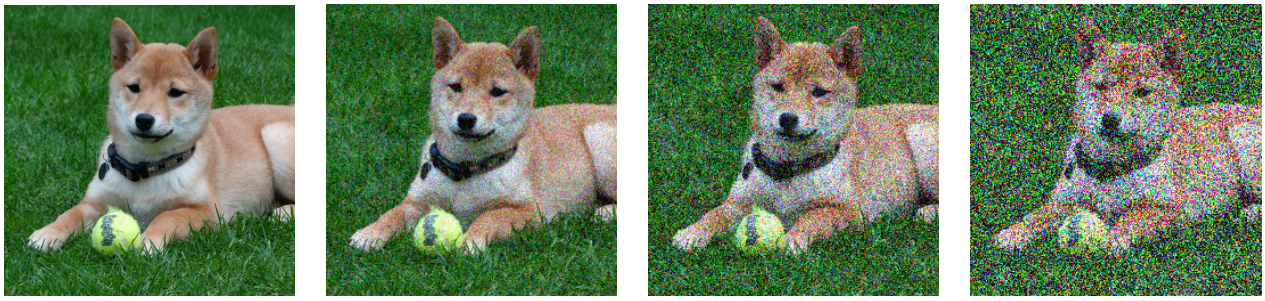
Data augmentation. During training, random cropping and random left-right flipping is applied to the images and their corresponding segmentation masks. We randomly crop the images to a fixed size of 1024×1024 for Cityscapes and 512×512 for ADE20K and Pascal Context. All of the denoising pretraining runs are conducted at a 512×512 resolution.

4.2. Decoder variants

We investigate the impact of decoder size on our results by varying the number of feature maps at the various stages of the decoder. The default ($1\times$) decoder configuration for all our experiments is $[1024, 512, 256, 128, 64]$ where the value at index i corresponds to the number of feature maps at the i^{th} decoder block. This is reflected in Figure 3. On Cityscapes, we experiment with doubling the default width of all decoder layers ($2\times$), while on Pascal Context and ADE20K, we experiment with tripling ($3\times$) the widths.

4.3. Training and inference

For downstream fine-tuning of the pretrained models for the semantic segmentation task, we use the standard pixel-wise cross-entropy loss. We use the Adam [50] optimizer with a cosine learning rate decay schedule. For both Denoising Pretraining (DeP) and Decoder Denoising Pretrain-



Original $\sigma = 0.1$ $\sigma = 0.2$ $\sigma = 0.4$

Figure 4. An illustration of a 256×256 image and a few reasonable values of standard deviation (σ) for Gaussian noise. For visualization, we clip noisy pixel values to [0, 1], but during training no clipping is used.

Table 1. Cityscapes mIoU on VAL_FINE set. Labeled examples are varied from full to 1/30 of the original TRAIN_FINE set

Method	Decoder width	full (2,975)	1/4 (744)	1/8 (372)	1/30 (100)
No Pretraining	1×	63.47	39.63	34.74	25.79
Supervised	1×	80.36	75.55	72.56	54.72
DeP	1×	77.14	68.87	63.90	53.41
DDeP	1×	**80.53**	**75.86**	**72.67**	**62.61**
No Pretraining	2×	62.25	37.72	33.73	24.93
Supervised	2×	80.50	75.57	72.84	60.36
DDeP	2×	**80.62**	**76.26**	**72.99**	**63.25**

ing (DDeP), we use a learning rate of 10^{-4} and a batch size of 128 and train for 6000 epochs.

The Pascal Context dataset is a subset of the larger Pascal VOC [27] dataset, and we find that denoising pretraining on all of Pascal VOC leads to a boost in downstream accuracy on Pascal Context. On ADE20K, denoising pretraining uses images from the TRAIN set, while on Cityscapes, we conduct denoising pretraining on the combination of TRAIN_FINE and TRAIN_COARSE sets.

When fine-tuning the pretrained models on the target semantic segmentation task, we sweep over weight decay and learning rate values between $[1\text{e}{-5}, 3\text{e}{-4}]$ and choose the best combination for each task. For the 100% setting, we report the means of 10 runs on all of the datasets. On Pascal Context and ADE20K, we also report the mean of 10 runs (with different subsets) for the 1%, 5% and 10% label fractions and 5 runs for the 20% setting. On Cityscapes, we report the mean of 10 runs for the $1/30$ setting, 6 runs for the $1/8$ setting and 4 runs for the $1/4$ setting.

During inference on Cityscapes, we evaluate on the full resolution 1024×2048 images by splitting them into two 1024×1024 input patches. We apply horizontal flip and average the results for each half. The two halves are concatenated to produce the full resolution output.

Table 2. Comparison with the state-of-the-art on Cityscapes. The result of [32] is reproduced by [108] based on DeepLab-v3+, while the results of [29, 44, 58, 63] are based on DeepLab-v2. All of the baselines except ours make use of a ResNet-101 backbone.

Method	full (2,975)	1/4 (744)	1/8 (372)	1/30 (100)
AdvSemSeg [44]	-	62.3	58.8	-
s4GAN [58]	65.8	61.9	59.3	-
DMT [29]	68.16	-	63.03	54.80
ClassMix [63]	-	63.63	61.35	-
CutMix [32]	-	68.33	65.82	55.71
PseudoSeg [108]	-	72.36	69.81	60.96
Sup. baseline [104]	74.88	73.31	68.72	56.09
PC^2Seg [104]	75.99	75.15	72.29	62.89
DDeP (Ours)	**80.62**	**76.26**	**72.99**	**63.25**

4.4. Performance gain by denoising pretraining

In Table 1, we report the results of DeP and DDeP on Cityscapes and compare them with the results of training from random initialization or initializing with an ImageNet-21K-pretrained encoder. DeP significantly outperforms randomly initialized models in all settings, and also outperforms the ImageNet-21K supervised baseline in the low label fraction settings. As shown in Figure 2, DeP outperforms the supervised baseline in the 1% and 2% labelled images settings. Decoder Denoising Pretraining (DDeP) further improves over both DeP and ImageNet-21K supervised pretraining for both the $1\times$ and $2\times$ decoder variants (Table 1).

Table 2 compares DDeP with previously proposed methods for label-efficient semantic segmentation on Cityscapes. DDeP outperforms these baseline methods at all training set sizes. With only 25% of the training data, DDeP produces better segmentations than the strongest baseline method, PC^2Seg [104], does when trained on the full dataset. Unlike most recent work, we do not perform evaluation at multiple scales, which should lead to further improvements.

Table 3. Pascal Context mIoU (%) on the VALIDATION set for labeled examples varied from 100% to 1% of the original TRAIN set. Supervised indicates ImageNet-21K pretraining of the backbone

Method	Decoder width	100% (4,998)	20% (1,000)	10% (500)	5% (250)	1% (50)
No pretraining	1×	17.64	8.32	6.50	5.20	2.67
Supervised	1×	59.50	42.05	35.02	28.24	11.93
DDeP	1×	**59.66**	**50.26**	**44.38**	**38.01**	**17.48**
No pretraining	3×	17.56	7.76	6.42	4.90	2.75
Supervised	3×	60.10	48.78	42.98	36.24	13.67
DDeP	3×	**60.13**	**53.64**	**49.95**	**44.30**	**23.20**

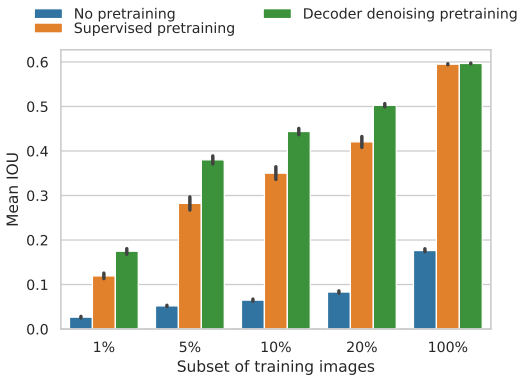


Figure 5. Performance on Pascal Context dataset (with a default 1× decoder) as a function of fraction of labeled data available.

On Pascal Context and ADE20K datasets, we evaluate DDeP on 1%, 5%, 10%, 20% and 100% of the training data, and obtain large improvements over supervised pretraining. Figure 5 compares the performance of DDeP with that of the supervised baseline and a randomly initialized model on Pascal Context. Table 3 compares these results with those obtained with a 3× decoder. Again, for both 1× and 3× decoders, DDeP significantly outperforms architecturally identical supervised models, obtaining improvements of 4-10% mIOU across all semi-supervised settings. Notably, with only 10% of the labels, DDeP outperforms the supervised model trained with 20% of the labels.

Figure 6 and Table 4 show the performance of DDeP on the ADE20K dataset. Again, we see gains of more than 10 points in the 5% and 10% settings and 5 points in the 1% setting. These consistent results demonstrate the effectiveness of DDeP across datasets and training set sizes.

Both the denoising pretrained and supervised ImageNet-21K pretrained models benefit from increasing the size of the decoder. However, this is not the case for the randomly initialized models. On all of the datasets, increasing the size of the decoder generally leads to a reduction in performance of training from random initialization (Tables 1, 3, 4). Our results emphasize the importance of pretraining for model scaling and the impact of finding the right pretraining strategy.

Table 4. ADE20K mIoU (%) on the VALIDATION set for labeled examples varied from 100% to 1% of the original TRAIN set. Supervised indicates ImageNet-21K pretraining of the backbone

Method	Decoder width	100% (20,210)	20% (4,042)	10% (2,021)	5% (1,010)	1% (202)
No pretraining	1×	20.32	6.35	4.55	3.30	2.10
Supervised	1×	48.43	39.40	22.25	10.05	4.85
DDeP	1×	**48.63**	**40.71**	**33.40**	**23.03**	**9.06**
No pretraining	3×	18.64	6.48	4.55	3.17	1.95
Supervised	3×	48.40	39.86	30.82	16.34	6.56
DDeP	3×	**48.92**	**41.17**	**36.14**	**28.49**	**13.23**

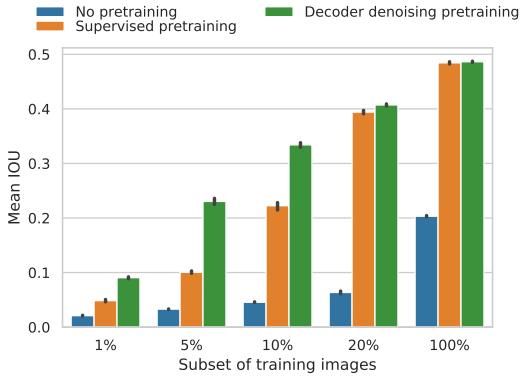


Figure 6. Performance on ADE20k dataset (with a default 1× decoder) as a function of fraction of labeled data available.

In Table 5, we train a standard U-Net with a ResNet50 encoder with DDeP on Pascal Context. DDeP outperforms the supervised baseline in all settings showing that our method generalizes beyond transformer architectures.

Table 5. Performance of a UNeT with a simple ResNet50 backbone on Pascal Context

Method	Decoder wd.	100%	20%	10%	5%	1%
No pretraining	1×	19.01	8.46	6.72	5.30	2.73
Supervised	1×	45.21	24.55	19.27	14.97	6.09
DDeP	1×	**46.07**	**30.38**	**26.39**	**21.12**	**9.63**

4.5. Ablation studies

We perform ablation studies on the Cityscapes and Pascal Context datasets to investigate the impact of various hyperparameters in the denoising pretraining setup.

4.5.1 Random noise levels

The denoising diffusion models, which inspire our work learn to reverse a diffusion process iteratively starting from pure white noise and ending at a clean image. In practice, these models are trained to denoise images corrupted with

Table 6. Comparison of fixed value of σ with uniform sampling of σ in the interval $[0.2, 0.3]$ on Pascal Context. Labeled examples are varied from 100% to 1% of the original TRAIN set, and mIoU (%) on the VALIDATION set is reported

Method	Decoder width	100% (4,998)	20% (1,000)	10% (500)
DDeP $\sigma \sim U(0.2, 0.3)$	3×	59.71	52.53	49.23
DDeP $\sigma = 0.2$	3×	**59.97**	**53.36**	**49.84**

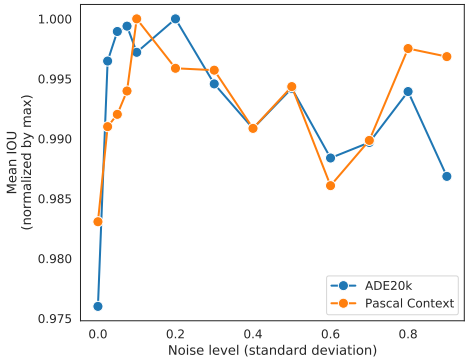


Figure 7. Effect of σ on downstream performance.

noise of a randomly sampled intensity according to a noise schedule. We therefore investigate this approach for representation learning. We randomize the choice of σ during pretraining and report the downstream semantic segmentation performance in Table 6. We use σ sampled uniformly at random between 0.2 and 0.3. Interestingly, we find that this strategy does not result in a performance boost in any of the settings, suggesting that simple denoising at a fixed noise scale is sufficient to learn useful image representations.

4.5.2 Impact of gamma

We pretrain the decoder on Pascal context and ADE20k and vary the value of σ. Figure 7 presents the results. We find that values of σ closer to 0 (less noisy images) perform better than higher values of σ. As shown in Figure 4, this level of noise is not very significant, but to perform this denoising task well, the model may still need to learn to identify object textures and edges.

4.5.3 Length of pretraining

Given the abundance of unlabeled data, in most settings, unsupervised pretraining is limited primarily by the time spent and the maximum quality of representations attainable with the method. We therefore test downstream performance as a function of pretraining duration and learning rate. We find that longer denoising pretraining consistently results in better downstream performance (Figure 8). This suggests that the performance of denoising pretraining is currently limited by time and compute, rather than an intrinsic limit of the quality of the learned representations. Accordingly,

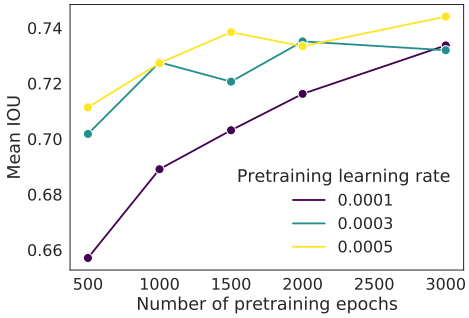


Figure 8. Effect of length of pretraining duration on downstream performance.

increasing the learning rate improves performance up to a point, but pretraining becomes unstable for excessive learning rates.

4.5.4 Denoising pretraining objective

In Eq. (4) our denoising objective function predicts the noise vector $\boldsymbol{\epsilon}$ rather than the noiseless image $\boldsymbol{x}$. We compare these two settings in Table 7 and find that predicting the noise is superior for both Pascal Context and ADE20K datasets.

Table 7. Compare noise prediction with image prediction

Method	Decoder width	Pascal Context	ADE20k
Predict x	1×	57.91	47.41
Predict ϵ	1×	**58.77**	**48.37**

5. Conclusion

Inspired by the recent popularity of diffusion probabilistic models for image synthesis, we investigate the effectiveness of these models in learning useful transferable representations for semantic segmentation. Surprisingly, we find that pretraining a semantic segmentation model as a denoising autoencoder leads to large gains in semantic segmentation performance, especially when the number of labeled examples is limited. We build on this observation and propose a two-stage pretraining approach in which supervised pretrained encoders are combined with denoising pretrained decoders. This leads to consistent gains across datasets and training set sizes, resulting in a practical approach to pretraining. Given these results, it is interesting to explore other types of corruption beyond simple Gaussian noise and study denoising pretraining on a single large and diverse unlabelled dataset of natural images. It is also interesting to explore the use of denoising pretraining for other dense prediction tasks.

References

[1] A. Agrawala. Learning with a probabilistic teacher. *IEEE Transactions on Information Theory*, 16(4):373–379, 1970. 2

[2] Anonymous. Label-efficient semantic segmentation with diffusion models. *Submitted to The Tenth International Conference on Learning Representations*, 2022. 2

[3] Relja Arandjelović and Andrew Zisserman. Object discovery with a copy-pasting gan. *arXiv:1905.11369*, 2019. 2

[4] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *arXiv:1906.00910*, 2019. 1, 3

[5] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv:2106.08254*, 2021. 2, 3

[6] Adam Bielski and Paolo Favaro. Emergence of object segmentation in perturbed generative models. *arXiv:1905.12663*, 2019. 2

[7] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with Transformers. *ECCV*, 2020. 3

[8] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. *ECCV*, 2018. 3

[9] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. *Semi-Supervised Learning*. MIT Press, 2006. 2

[10] Jieneng Chen, Yongyi Lu, Qihang Yu, Xiangde Luo, Ehsan Adeli, Yan Wang, Le Lu, Alan L. Yuille, and Yuyin Zhou. Transunet: Transformers make strong encoders for medical image segmentation. *CoRR*, abs/2102.04306, 2021. 3

[11] Liang-Chieh Chen, Raphael Gontijo Lopes, Bowen Cheng, Maxwell D Collins, Ekin D Cubuk, Barret Zoph, Hartwig Adam, and Jonathon Shlens. Naive-student: Leveraging semi-supervised learning in video sequences for urban scene segmentation. *ECCV*, 2020. 3

[12] 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. *TPAMI*, 40(4):834–848, 2017. 5

[13] Mickaël Chen, Thierry Artières, and Ludovic Denoyer. Unsupervised object segmentation by redrawing. *arXiv:1905.13539*, 2019. 2

[14] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. *International Conference on Machine Learning*, pages 1691–1703, 2020. 2

[15] Nanxin Chen, Yu Zhang, Heiga Zen, Ron J Weiss, Mohammad Norouzi, and William Chan. WaveGrad: Estimating gradients for waveform generation. *International Conference on Learning Representations*, 2021. 2

[16] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *International conference on machine learning*, pages 1597–1607, 2020. 1, 3

[17] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv:2003.04297*, 2020. 3

[18] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, M. Enzweiler, Rodrigo Benenson, Uwe Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016. 5

[19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805*, 2018. 2

[20] J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *NAACL*, 2019. 3

[21] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv:1708.04552*, 2017. 2

[22] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *ArXiv*, abs/2105.05233, 2021. 2, 3

[23] Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1422–1430, 2015. 3

[24] Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. *arXiv:1907.02544*, 2019. 2

[25] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 3

[26] A. Dosovitskiy, L. Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, M. Dehghani, Matthias Minderer, G. Heigold, S. Gelly, Jakob Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2021. 3, 4

[27] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010. 6

[28] Zhengyang Feng, Qianyu Zhou, Guangliang Cheng, Xin Tan, Jianping Shi, and Lizhuang Ma. Semi-supervised semantic segmentation via dynamic self-training and class-balanced curriculum. *arXiv:2004.08514*, 2020. 3

[29] Zhengyang Feng, Qianyu Zhou, Qiqi Gu, Xin Tan, Guangliang Cheng, Xuequan Lu, Jianping Shi, and Lizhuang Ma. DMT: Dynamic mutual training for semi-supervised learning. *arXiv:2004.08514*, 2020. 2, 6

[30] S Fralick. Learning to recognize patterns without a teacher. *IEEE Transactions on Information Theory*, 1967. 2

[31] Geoff French, Samuli Laine, Timo Aila, Michal Mackiewicz, and Graham Finlayson. Semi-supervised semantic segmentation needs strong, varied perturbations. *BMVC*, 2019. 2, 3

[32] Geoffrey French, Samuli Laine, Timo Aila, Michal Mackiewicz, and Graham Finlayson. Semi-supervised semantic segmentation needs strong, varied perturbations. *BMVC*, 2020. 2, 6

[33] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. *CVPR*, 2021. 2

[34] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv:1803.07728*, 2018. 3

[35] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *NeurIPS*, pages 2672–2680, 2014. 2

[36] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv:2006.07733*, 2020. 1, 3

[37] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020. 1, 3

[38] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CVPR*, 2016. 1

[39] Olivier J Hénaff, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. *arXiv:1905.09272*, 2019. 3

[40] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv:1808.06670*, 2018. 1, 3

[41] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv:2006.11239*, 2020. 2, 3, 4, 5

[42] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *arXiv:2106.15282*, 2021. 2, 3

[43] Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. Argmax flows and multinomial diffusion: Towards non-autoregressive language models. *arXiv:2102.05379*, 2021. 2

[44] Wei Chih Hung, Yi Hsuan Tsai, Yan Ting Liou, Yen-Yu Lin, and Ming Hsuan Yang. Adversarial learning for semi-supervised semantic segmentation. *BMVC*, 2018. 2, 6

[45] Wei-Chih Hung, Yi-Hsuan Tsai, Yan-Ting Liou, Yen-Yu Lin, and Ming-Hsuan Yang. Adversarial learning for semi-supervised semantic segmentation. *BMVC*, 2018. 2

[46] Aapo Hyvärinen and Peter Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005. 3

[47] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 2

[48] Zhanghan Ke, Kaican Li Di Qiu, Qiong Yan, and Rynson WH Lau. Guided collaborative training for pixel-wise semi-supervised learning. *ECCV*, 2020. 2

[49] Jongmok Kim, Jooyoung Jang, and Hyunwoo Park. Structured consistency loss for semi-supervised semantic segmentation. *arXiv:2001.04647*, 2020. 3

[50] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2015. 5

[51] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 491–507, 2020. 1

[52] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. *CVPR*, 2019. 3

[53] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. *European conference on computer vision*, pages 577–593, 2016. 3

[54] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv:1910.13461*, 2019. 2

[55] Haoying Li, Yifan Yang, Meng Chang, Huajun Feng, Zhihai Xu, Qi Li, and Yueting Chen. Srdiff: Single image super-resolution with diffusion probabilistic models. *arXiv:2104.14951*, 2021. 3

[56] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv:1907.11692*, 2019. 2, 3

[57] 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. *arXiv preprint*, 2021. 3

[58] Sudhanshu Mittal, Maxim Tatarchenko, and Thomas Brox. Semi-supervised semantic segmentation with high-and low-level consistency. *TPAMI*, 43(4):1369–1379, 2021. 2, 6

[59] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: A regularization method for supervised and semi-supervised learning. *TPAMI*, 41(8):1979–1993, 2018. 2

[60] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Loddon Yuille. The role of context for object detection and semantic segmentation in the wild. *2014 IEEE*

Conference on Computer Vision and Pattern Recognition, pages 891–898, 2014. 5

[61] Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. *arXiv:2102.09672*, 2021. 2, 3

[62] M. Noroozi and P. Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. *ECCV*, 2016. 3

[63] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. Classmix: Segmentation-based data augmentation for semi-supervised learning. *WACV*, 2021. 2, 6

[64] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018. 1, 3

[65] Yassine Ouali, Céline Hudelot, and Myriam Tami. Semi-supervised semantic segmentation with cross-consistency training. *CVPR*, 2020. 2, 3

[66] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2536–2544, 2016. 3

[67] Pedro H. O. Pinheiro, Amjad Almahairi, Ryan Y. Benmaleck, Florian Golemo, and Aaron C. Courville. Unsupervised learning of dense visual representations. *ArXiv*, abs/2011.05499, 2020. 3

[68] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv:2103.00020*, 2021. 1

[69] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv:1910.10683*, 2019. 2

[70] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 2020. 3

[71] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *arXiv:2102.12092*, 2021. 3

[72] Tal Remez, Jonathan Huang, and Matthew Brown. Learning to segment via cut-and-paste. *ECCV*, 2018. 2

[73] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *MICCAI*, 2015. 2, 3

[74] Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. Semi-supervised self-training of object detection models. *Applications of Computer Vision and the IEEE Workshop on Motion and Video Computing, IEEE Workshop on*, 1:29–36, 2005. 3

[75] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David J Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. *arXiv:2111.05826*, 2021. 3

[76] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, D. Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *ArXiv*, abs/2104.07636, 2021. 2, 3

[77] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *NeurIPS*, 29:2234–2242, 2016. 2

[78] H Scudder. Probability of error of some adaptive pattern-recognition machines. *IEEE Transactions on Information Theory*, 1965. 2

[79] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 806–813, 2014. 1

[80] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *International Conference on Machine Learning*, pages 2256–2265, 2015. 2, 3

[81] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *NeurIPS*, pages 11895–11907, 2019. 2, 3

[82] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *International Conference on Learning Representations*, 2021. 3

[83] Nasim Souly, Concetto Spampinato, and Mubarak Shah. Semi supervised semantic segmentation using generative adversarial network. *ICCV*, 2017. 2

[84] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. *arXiv:2105.05633*, 2021. 3

[85] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015. 2

[86] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *NeurIPS*, 2017. 2

[87] Jesper E Van Engelen and Holger H Hoos. A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440, 2020. 2

[88] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, pages 5998–6008, 2017. 3

[89] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011. 3, 5

[90] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. *ICML*, 2008. 3

[91] Pascal Vincent, H. Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. *ICML '08*, 2008. 4

[92] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol, and Léon Bottou. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *JMLR*, 2010. 3, 4

[93] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan L. Yuille, and Liang-Chieh Chen. Max-deeplab: End-to-end panoptic segmentation with mask transformers. *arXiv preprint*, 2020. 3

[94] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-Deeplab: Stand-alone axial-attention for panoptic segmentation. *ECCV*, 2020. 3

[95] Xiaolong Wang, Ross B. Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. *CVPR*, 2018. 3

[96] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. *CVPR*, 2021. 3

[97] David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. *33rd annual meeting of the association for computational linguistics*, pages 189–196, 1995. 2

[98] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. *CVPR*, 2019. 2

[99] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. *ECCV*, 2016. 3

[100] Richard Zhang, Phillip Isola, and Alexei A. Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 645–654, 2017. 3

[101] 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. *CVPR*, 2021. 2

[102] Xiangyu Zhao, Raviteja Vemulapalli, P. A. Mansfield, Boqing Gong, Bradley Green, L. Shapira, and Ying Wu. Contrastive learning for label-efficient semantic segmentation. *ArXiv*, abs/2012.06985, 2020. 5

[103] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H.S. Torr, and Li Zhang. Rethinking semantic segmentation from a sequence-to-sequence perspective with Transformers. *arXiv preprint*, 2020. 3

[104] Yuanyi Zhong, Bodi Yuan, Hong Wu, Zhiqiang Yuan, Jian Peng, and Yu-Xiong Wang. Pixel contrastive-consistent semi-supervised semantic segmentation. *ICCV*, 2021. 2, 3, 6

[105] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127:302–321, 2018. 5

[106] Yi Zhu, Zhongyue Zhang, Chongruo Wu, Zhi Zhang, Tong He, Hang Zhang, R Manmatha, Mu Li, and Alexander Smola. Improving semantic segmentation via self-training. *arXiv:2004.14960*, 2020. 3

[107] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D Cubuk, and Quoc V Le. Rethinking pre-training and self-training. *arXiv:2006.06882*, 2020. 2, 3

[108] Yuliang Zou, Zizhao Zhang, Han Zhang, Chun-Liang Li, Xiao Bian, Jia-Bin Huang, and Tomas Pfister. PseudoSeg: Designing pseudo labels for semantic segmentation. *ICLR*, 2021. 2, 3, 6