Unleashing Text-to-Image Diffusion Models for Visual Perception

Wenliang Zhao[1]* Yongming Rao[1]* Zuyan Liu[1]* Benlin Liu[2] Jie Zhou[1] Jiwen Lu[1]†

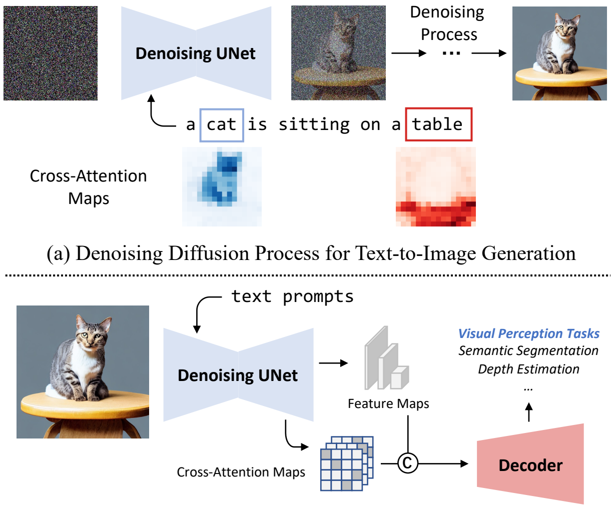
[1]Tsinghua University [2]University of Washington

Abstract

Diffusion models (DMs) have become the new trend of generative models and have demonstrated a powerful ability of conditional synthesis. Among those, text-to-image diffusion models pre-trained on large-scale image-text pairs are highly controllable by customizable prompts. Unlike the unconditional generative models that focus on low-level attributes and details, text-to-image diffusion models contain more high-level knowledge thanks to the vision-language pre-training. In this paper, we propose VPD (Visual Perception with a pre-trained Diffusion model), a new framework that exploits the semantic information of a pre-trained text-to-image diffusion model in visual perception tasks. Instead of using the pre-trained denoising autoencoder in a diffusion-based pipeline, we simply use it as a backbone and aim to study how to take full advantage of the learned knowledge. Specifically, we prompt the denoising decoder with proper textual inputs and refine the text features with an adapter, leading to a better alignment to the pre-trained stage and making the visual contents interact with the text prompts. We also propose to utilize the cross-attention maps between the visual features and the text features to provide explicit guidance. Compared with other pre-training methods, we show that vision-language pre-trained diffusion models can be faster adapted to downstream visual perception tasks using the proposed VPD. Extensive experiments on semantic segmentation, referring image segmentation and depth estimation demonstrates the effectiveness of our method. Notably, VPD attains 0.254 RMSE on NYUv2 depth estimation and 73.3% oIoU on RefCOCO-val referring image segmentation, establishing new records on these two benchmarks. Code is available at https://github.com/wl-zhao/VPD.

1. Introduction

Recently, large text-to-image diffusion models [43, 40] have demonstrated phenomenal power in generating diverse and high-fidelity images with high customizability [43, 19, 36, 6], attracting growing attention from both the research community and the public eye. By leveraging large-scale datasets of image-text pairs (*e.g.*, LAION-5B [46]), text-to-image diffusion models exhibit favorable scaling ability. Large-scale text-to-image diffusion models are able to generate high-quality images with rich texture, diverse content and reasonable structures while having compositional and editable semantics. This phenomenon potentially suggests that large text-to-image diffusion models can *implicitly* learn both high-level and low-level visual concepts from massive image-text pairs. Moreover, recent research [19, 36] also has highlighted the clear correlations between the latent visual features and corresponding words in text prompts in text-to-image diffusion models.



(a) Denoising Diffusion Process for Text-to-Image Generation

(b) **VPD**: Visual Perception with Pre-trained Diffusion Models

Figure 1. **The main idea of the proposed VPD framework.** Motivated by the compelling generative semantic of a text-to-image diffusion model, we proposed a new framework named VPD to exploit the pre-trained knowledge in the denoising UNet to provide semantic guidance for downstream visual perception tasks.

The compelling generative semantic and compositional abilities of text-to-image diffusion models motivate us to think: *is it possible to extract the visual knowledge learned by large diffusion models for visual perception tasks?*

*Equal contribution. †Corresponding authors.

However, it is non-trivial to solve this problem. Conventional visual pre-training methods aim to encode the input image as latent representations and learn the representations with pretext tasks like contrastive learning [18, 10] and masked image modeling [2, 17] or massive annotations in classification and vision-language tasks. The pre-training process makes the learned latent representation naturally suitable for a range of visual perception tasks as semantic knowledge is extracted from the raw images. In contrast, text-to-image models are designed to generate high-fidelity images based on textual prompts. Text-to-image diffusion models take as input random noises and text prompts, and aim to produce images through a progressive denoising process [43, 20]. While there is a notable gap between the text-to-image generation task and the conventional visual pre-training mechanisms, the training process of text-to-image models also requires them to capture both low-level knowledge of images (*e.g*., textures, edge, and structures) and high-level semantic relations between visual and linguistic concepts from diverse and large-scale image-text pairs in an implicit way. Although rich representations are learned in large diffusion models, it is still unknown how to extract this knowledge for various visual perception tasks and whether it can benefit visual perception.

In this paper, we study how to leverage the knowledge learned in text-to-image for visual perception. Compared to transferring knowledge from conventional pre-trained models to downstream visual perception tasks, there are two distinct challenges to performing transfer learning on diffusion models: the incompatibility between the diffusion pipeline and visual perception tasks and the architectural differences between UNet [44]-like diffusion models and popular visual backbones. To tackle these challenges, we introduce a new framework called *VPD* to adapt pre-trained diffusion models for visual perception tasks. Instead of using the step-by-step diffusion pipeline, we propose to simply employ the autoencoder as a backbone model to directly consume the natural images without noise and perform a single extra denoising step with designed prompts to extract the semantic information. Our framework is based on popular Stable Diffusion [43] models, which conduct the denoising process in a learned latent space with a UNet architecture. We extract features from different hierarchies from the UNet decoder to construct visual representations of the input image. To align with the pre-trained stage and facilitate interactions between visual content and text prompts, we prompt the denoising diffusion model with proper textual inputs and refine the text features with an adapter. Additionally, inspired by previous studies on the relations between prompt words and visual patterns in diffusion models, we propose to utilize the cross-attention maps between the visual and text features to provide explicit guidance. The combined implicit and explicit guidance can be fed to various visual decoders to perform visual perception tasks. Our main idea is summarized in Figure 1.

We evaluate our method on three representative visual perception tasks covering: 1) semantic segmentation [58] which requires the understanding of high-level and fine-grained visual concepts, 2) referring image segmentation [56, 33] that requires the ability of visual-language modeling, and 3) depth estimation [47] that requires low-level and structural knowledge of images. With the help of the proposed VPD, we show that a vision-language pre-trained diffusion model can be a fast and powerful learner of downstream visual perception tasks. Our method attains 73.3% oIoU and 0.254 RMSE on RefCOCO [56] referring image segmentation and NYUv2 [47] depth estimation, respectively, establishing new state-of-the-art on these two benchmarks. Equipped with a lightweight Semantic FPN [24] decoder, our model achieves 54.6% mIoU on ADE20K [58], outperforming supervisedly pre-trained ConvNeXt-XL [29] model with comparable computational complexity. We also exhibit that models pre-trained with diffusion tasks can fast obtain 44.7% mIoU on this challenging benchmark with only 4K iteration training, outperforming existing pre-training methods. We expect our study to offer a new perspective on learning more generic visual representations with generative models and spark further research on bridging and unifying the vibrant research fields of image generation and perception.

2. Related Work

Diffusion Models. Diffusion denoising probabilistic models, also known as diffusion models, have emerged as a new prevailing family of generative models that demonstrate remarkable synthesis quality and controllability. The fundamental concept behind the diffusion model involves training a denoising autoencoder to learn the inverse of a Markovian diffusion process [48, 20]. With proper reparameterization, the training objective of diffusion models can be formulated as a simple weighted MSE loss [20], which makes diffusion models enjoy more stable training compared with GANs [16] and VAEs [23]. Sampling from a diffusion model [49, 26, 30] can then be viewed as a progressive denoising procedure, which requires multiple evaluations of the denoising autoencoder. As a step towards high-resolution image synthesis based on diffusion models, Rombach *et al*. [43] propose the latent diffusion models (LDMs), which perform diffusion on a latent space of a lower resolution and thus can significantly reduce the computational costs. They also propose a generic solution to add conditions via the cross-attention [51] mechanism. These advancements allow for training text-to-image diffusion models on a large-scale dataset LAION-5B [46], which are now available in the famous “Stable-Diffusion”

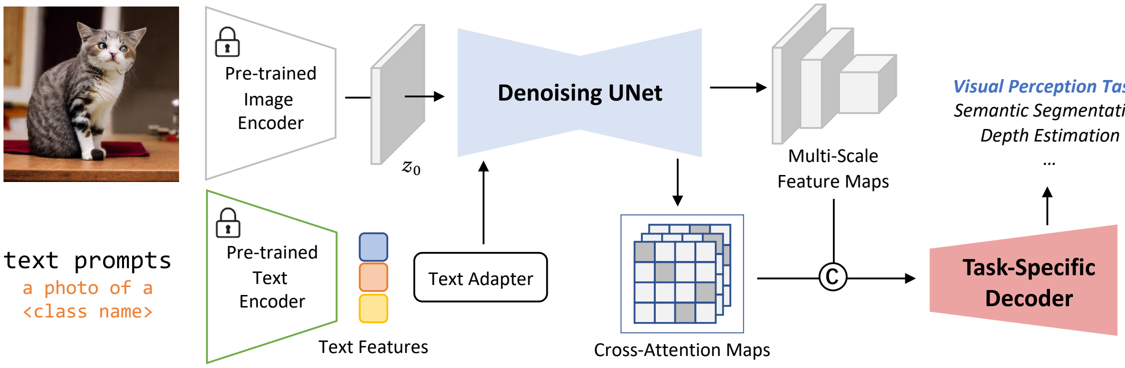


Figure 2. **The overall framework of VPD.** To better exploit the semantic knowledge learned from text-to-image generation pre-training, we prompt the denoising UNet with properly designed text prompts and employ the cross-attention maps to provide both implicit and explicit guidance to downstream visual perception tasks. Our framework can fully leverage both the low-level and high-level pre-trained knowledge and can be applied in a variety of visual perception tasks.

library. Recent work by [19] has witnessed a clear visual-text correlation in the large text-to-image diffusion models, which motivates us to study whether the pre-trained knowledge can be exploited to facilitate downstream visual perception tasks. Different from previous diffusion-based framework [9, 1] that reformulate the visual perception task as progressive denoising, we employ the denoising autoencoder pre-trained on the text-to-image generation as a backbone and study how to make full use of the learned high-level and low-level knowledge, which only require a single forward pass of the denoising autoencoder.

Visual Pre-training. The pre-training & fine-tuning paradigm has significantly pushed the development of computer vision, especially in downstream visual perception tasks where labels are hard to collect. The most widely used pre-training is supervised pre-training on large-scale image classification datasets like ImageNet [12]. Besides, self-supervised learning such as contrastive learning [7, 18] and masked image modelling [38, 17] have also proved to be able to learn transferrable representations. In this paper, we will demonstrate that large-scale text-to-image generation can also be a possible alternative for visual pre-training. Different from the standard visual pre-training methods that are specifically designed for extracting high-level representation of visual data, a model trained on a generative task focuses on the synthesis quality and captures more low-level clues. However, our results show that due to the existence of natural language during pre-training, a well-learned text-to-image diffusion model contains sufficient both high-level and low-level knowledge, which can also be applied in downstream visual perception tasks.

3. Method

In this section, we present VPD, a new framework that achieves visual perception with a pre-trained diffusion model. Our key idea is to investigate how to fully extract the pre-trained high-level knowledge in a pre-trained text-to-image diffusion model. We will start by reviewing the background of diffusion models, and then describe our designs of VPD, including how to implicitly and explicitly leverage the visual-language correspondence lies in the pre-trained text-to-image diffusion models. The overall framework of our VPD is illustrated in Figure 2.

3.1. Preliminaries: Diffusion Models

To begin with, we will provide a brief overview of the diffusion models [48, 20, 22]. Diffusion models are a new family of generative models that can reconstruct the distribution of data by learning the reverse process of a diffusion process. Denoting $\boldsymbol{z}_t$ as the random variable at t-th timestep, the diffusion process is modeled as a Markov:

$$\boldsymbol{z}_t \sim \mathcal{N}(\sqrt{\alpha_t}\boldsymbol{z}_{t-1}, (1-\alpha_t)\boldsymbol{I}), \tag{1}$$

where $\{\alpha_t\}$ are fixed coefficients that determine the noise schedule. The above definition leads to a simple close form of $p(\boldsymbol{z}_t|\boldsymbol{z}_0)$:

$$\begin{aligned} \boldsymbol{z}_t &= \sqrt{\bar{\alpha}_t}\boldsymbol{z}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon} \\ \bar{\alpha}_t &= \prod_{s=1}^{t} \alpha_s, \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \end{aligned} \tag{2}$$

which further allows sampling an arbitrary $\boldsymbol{z}_t$ efficiently during training. With proper re-parameterization, the training objective of diffusion models can be derived as [20]:

$$L_{\rm DM} = \mathbb{E}_{\boldsymbol{z}_0,\boldsymbol{\epsilon},t}\left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_t(\boldsymbol{z}_0,\boldsymbol{\epsilon}), t; \mathcal{C})\|_2^2\right], \tag{3}$$

where $\boldsymbol{z}_t$ is computed as Equation (2). $\boldsymbol{\epsilon}_\theta$ is an autoencoder (usually implemented as a UNet [44]) that is learned to predict the $\boldsymbol{\epsilon}$ given the conditioning inputs $\mathcal{C}$. The sampling of diffusion models is achieved by discretizing the diffusion SDE or ODE [50] thus requires multiple model evaluations at different timesteps.

The training objective (3) enables stable training of diffusion models, even with complex conditioning inputs. Recently, [43] released a text-to-image model (namely "Stable-Diffusion") trained on large-scale image-text dataset LAION-5B [46], which has demonstrated remarkable performance on image synthesis controlled by natural language. Specifically, they first train a VQGAN consisting of an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$, which can achieve the conversion between the pixel space and the latent space. They then train a diffusion model on that latent space with the same objective in Equation (3). In this work, we will exploit how to fully use the learned high-level knowledge of the pre-trained text-to-image diffusion model in the downstream visual tasks.

3.2. Prompting Text-to-Image Diffusion Model

A pre-trained diffusion model contains sufficient information to sample from the data distribution since the model $\boldsymbol{\epsilon}_\theta$ can be viewed as the learned gradient of data density $\nabla_{\boldsymbol{z}_t} \log p(\boldsymbol{z}_t|\mathcal{C})$ [3]. As for the text-to-image model, we believe that there is enough high-level knowledge due to the weak supervision of the natural language during pre-training. Our goal is to fully exploit the knowledge of a well-trained text-conditioned $\boldsymbol{\epsilon}_\theta$ and transfer the learned knowledge to downstream visual perception tasks. A general perception task aims to model the distribution $p(\boldsymbol{y}|\boldsymbol{x})$, where $\boldsymbol{y}$ is the task-specific label and $\boldsymbol{x}$ is the input image. Our basic idea is to build a connection between the task-specific label and the natural language, such that the learned semantic information can be efficiently extracted. To achieve this, we first rewrite the prediction model as $p_\phi(\boldsymbol{y}|\boldsymbol{x}, \mathcal{S})$, where $\mathcal{S}$ is a set containing all the category names of the task. This is reasonable since the label $\boldsymbol{y}$ is related to the $\mathcal{S}$ in both shape and semantic meaning. We then implement $p_\phi(\boldsymbol{y}|\boldsymbol{x}, \mathcal{S})$ as:

$$p_\phi(\boldsymbol{y}|\boldsymbol{x}, \mathcal{S}) = p_{\phi_3}(\boldsymbol{y}|\mathcal{F}) p_{\phi_2}(\mathcal{F}|\boldsymbol{x}, \mathcal{C}) p_{\phi_1}(\mathcal{C}|\mathcal{S}), \quad (4)$$

where $\mathcal{F}$ is a set of feature maps and $\mathcal{C}$ denotes the text features. We now describe each term in Equation (4) and its instantiation in detail:

(1) $\underline{p_{\phi_1}(\mathcal{C}|\mathcal{S})}$ is responsible to extract text features from the class names. We use the same CLIP [39] text encoder as the pre-training stage of Stable-Diffusion [43], and the text inputs are simply defined using a template of "`a photo of a [CLS]`". However, the domain gap is usually witnessed when transferring the text encoder to downstream tasks [59, 15]. Inspired by [15], we use a text adapter implemented as a two-layer MLP to refine the text features obtained by the CLIP text encoder. To sum up, the text features are computed as follows:

$$\begin{aligned}
&\mathcal{T} \leftarrow \{\text{template}(s)|s \in \mathcal{S}\} \\
&\mathcal{C} \leftarrow \text{CLIPTextEncoder}(\mathcal{T}) \\
&\mathcal{C} \leftarrow \mathcal{C} + \gamma \,\text{MLP}(\mathcal{C}),
\end{aligned} \quad (5)$$

where $\mathcal{T}$ denotes the raw texts generated by applying the prompt template to the set of class names and γ is a learnable scale factor that is initialized to be very small (*e.g.*, 1e-4). This design can help us to maximally preserve the pre-trained knowledge of the text encoder, as well as mitigate the domain gap between the pre-training task and the downstream task. Note that different from the usage of CLIP text encoder in [43] where the features of the whole sentence are used, we simply use the feature from the `[EOS]` token. Therefore, the shape of $\mathcal{C}$ is $|\mathcal{S}| \times C$ where C is the output dimension of the CLIP text encoder.

(2) $\underline{p_{\phi_2}(\mathcal{F}|\boldsymbol{x}, \mathcal{C})}$ aims to extract hierarchical feature maps $\mathcal{F}$ given the input image $\boldsymbol{x}$ and the conditioning inputs $\mathcal{C}$. Since $\mathcal{C}$ contains information from the natural language, p_{ϕ_2} needs to capture the cross-domain interactions between vision and language. Interestingly, we find the pre-trained text-to-image diffusion model can be a very good initialization of p_{phi_2}. Although $\boldsymbol{\epsilon}_\theta$ is trained to perform score-matching [50] according to the training objective, it has already bridged the vision and language domains. In our implementation, we first use the encoder of the VQGAN $\mathcal{E}$ to encode the image into the latent space (*e.g.*, $\boldsymbol{z}_0 = \mathcal{E}(\boldsymbol{x})$) and then feed the latent feature map and the conditioning inputs to the pre-trained $\boldsymbol{\epsilon}_\theta$ network. Note that we simply set $t = 0$ such that no noise is added to the latent feature map. The hierarchical features $\mathcal{F}$ can also be easily obtained from the last layer of each output block in different resolutions. Typically, the size of the input image is 512×512 and $\mathcal{F}$ contains 4 feature maps, where the i-th feature map F_i has the spatial size of $H_i = W_i = 2^{i+2}$, $i = 1, 2, 3, 4$.

(3) $\underline{p_{\phi_3}(\boldsymbol{y}|\mathcal{F})}$ is the prediction head that generates results from the hierarchical feature maps $\mathcal{F}$. We implement p_{ϕ_3} as a Semantic FPN [24], consisting of several convolutional layers and upsampling layers. The prediction head can be designed to be very lightweight since the $\boldsymbol{\epsilon}_\theta$ already has enough capacity to perform downstream vision tasks.

The above formulation enables us to decompose the general visual perception tasks such that the role of the pre-trained diffusion model can be better understood. By injecting the task-specific labels $\mathcal{S}$ as the inputs, we implicitly prompt the pre-trained denoising autoencoder to explore the learned semantic knowledge. It is also worth noting that our method is not a diffusion-based framework anymore, because we only use a single UNet as a backbone (see Figure 1 to better understand the differences).

3.3. Semantic Guidance via Cross-attention

Apart from designing proper prompts to implicitly extract high-level knowledge from ϵ_θ network, we also propose to use the cross-attention map as an explicit semantic guidance. It has been observed in [19] that in a well-trained text-to-image diffusion model, the cross-attention map between the feature map and the conditioning text feature enjoys good locality. This nice property motivates us to leverage the cross-attention maps to explicitly facilitate downstream visual perception. The cross-attention operation exists in each of the 4 resolutions of the ϵ_θ network. Therefore, for the i-th resolution, we can simply average all the cross-attention maps belonging to the resolution to obtain an averaged map A_i. Since the cross-attention maps are computed by using the conditioning inputs $\mathcal{C}$ as the key and value, the averaged attention map has the shape of $A_i \in \mathbb{R}^{|S| \times H_i \times W_i}$.

The averaged cross-attention map is useful because each channel of it aggregates some semantic information of a certain category. We can then concatenate the averaged cross-attention maps with the original hierarchical feature maps and fed the results to the prediction head, *i.e.*, $F_i \leftarrow [F_i, A_i]$. By default, we do not use the cross-attention maps at the lowest resolution since they are not very accurate (which we will analyze in the experiments). We empirically find that explicit semantic guidance through cross-attention can help our model faster adapt to downstream tasks.

3.4. Implementation

We consider three visual perception tasks in this work, including semantic segmentation, referring image segmentation, and depth estimation. Basically, we use a similar architecture for these tasks, as mentioned above. However, there are some differences in minor design, which we will describe as follows. Firstly, the procedure to obtain the conditioning inputs $\mathcal{C}$ slightly differs in different tasks. For semantic segmentation, $\mathcal{S}$ contains the class names in the dataset. For referring image segmentation, we simply use the referring expression (a single sentence) to compute the conditioning inputs $\mathcal{C}$. For depth estimation, we can build the text prompt similarly using the category name of the scene, such as "kitchen", "bathroom", *etc*. Second, the output channels of the task-specific head $p_{\phi_3}(\boldsymbol{y}|\mathcal{F})$ are different. Third, the training objective of the three tasks are varied. We use the cross-entropy loss for both semantic segmentation and referring image segmentation, while the Scale-Invariant loss (SI) [14] is used for depth estimation.

4. Experiments

To verify the effectiveness of our method, we conduct experiments on three visual tasks including referring image segmentation, semantic segmentation, and depth estimation, covering both high-level and low-level visual perception. We will first present the experimental settings of these tasks and then give our main results. We will also provide detailed ablation studies and analyses of our method.

4.1. Experiment Setups

We first provide some common configurations of VPD. For all three downstream tasks, we fix the VQGAN encoder $\mathcal{E}$ and the CLIP text encoder during training. To fully preserve the pre-trained knowledge of the ϵ_θ, we always set the learning rate of ϵ_θ as 1/10 of the base learning rate. We use γ=1e-4 for the text adapter. The task-specific settings and training details are elaborated as follows.

Semantic Segmentation. The goal of semantic segmentation is to assign pixel-level labels to a given image, which requires a fine-grained high-level understanding of visual content. We evaluate our method on ADE20K [58], which consists of 20K images for training and 2K images for validation. Since our method can adapt faster to the downstream tasks, we train our model for 80K iterations using a Semantic FPN [24] by default. We use a global batch size of 16 and set the learning rate as 1e-4. We use the AdamW optimizer with a weight decay of 1e-4 and warming-up iterations of 1500. We adopt the polynomial learning rate scheduler with a power of 0.9 and a minimum learning rate of 1e-6. For the fast schedule (8K iterations), we linear scale the learning rate schedule and set the warming-up iterations to 150. During inference, we use the slide inference with a crop size 512 × 512 and a stride of 341 × 341.

Referring Image Segmentation. Referring image segmentation aims to find the related object given a natural language expression from an image. We perform experiments on the widely used benchmark RefCOCO [56], RefCOCO+ [56], and more challenging G-Ref [34] datasets with significantly longer expressions. RefCOCO, RefCOCO+, and G-Ref contain around 20K images and 50K annotated objects, with 142,209, 141,564, and 104,560 annotated expressions respectively. Following common practice, we train our model on the training set and evaluate the validation set. We use the overall intersection-over-union (oIoU) as the metric to compare different methods. As for the decoder head, we follow LAVT [54] which uses a simple convolution head to fusion the features and generate the semantic prediction. We train our model for 40 epochs with a total batch size of 32. We set the learning rate as 5e-5 and the weight decay as 0.01. As we have multiple expressions on a single image, during the training phase, we randomly choose a language description. In the inference time, we evaluate sequentially and calculate the mean results following common practice.

Depth Estimation. We adopt a widely used benchmark NYUv2 [47] to evaluate our method in depth estimation.

Table 1. **Semantic segmentation with different methods.** We compare our VPD with previous methods including supervised pre-training and self-supervised pre-training. While other methods adopt the UPerNet [52] segmentation head, we find our VPD can achieve good results with a more lightweight Semantic FPN [24] with smaller crop size and fewer training iterations.

Method	#Iters	Crop	FLOPs	mIoU^{ss}	mIoU^{ms}
supervised pre-training					
Swin-L [28]	160K	640^2	647G	52.1	53.5
ConvNeXt-L [29]	160K	640^2	614G	53.2	53.7
ConvNeXt-XL [29]	160K	640^2	834G	53.6	54.0
self-supervised pre-training					
MAE-ViT-L/16 [17]	126K	-	-	53.6	-
visial-language pre-training					
CLIP-ViT-B [42]	80K	640^2	340G	50.6	51.3
text-to-image pre-training					
VPD (Ours)	80K	512^2	891G	**53.7**	**54.6**

NYUv2 contains 24K images for training and 645 images for testing, covering 464 indoor scenes. Following common practice, we report the absolute relative error (REL), root mean squared error (RMSE), and average log10 error between predicted depth $\hat{d}$ and the ground truth depth d. We also report the threshold accuracy δ_n which denotes $\delta_n = \%$ of pixels satisfying $\max(d_i/\hat{d}_i, \hat{d}_i/d_i) < 1.25^n$ for $n = 1, 2, 3$. During training, we randomly crop the images to 480×480. We set the learning rate as 5e-4 and train the model for 25 epochs with batch size of 24. The decoder head and other experimental setting is the same as [53]. We use the flip and sliding windows during testing.

4.2. Main Results

In this section, we will provide our main results on three downstream tasks, including semantic segmentation, referring image segmentation, and depth estimation. Apart from training the models using the default schedule, we also perform experiments on a faster scheduler with very few iterations or epochs to show that our method can quickly adapt to downstream visual perception tasks.

Semantic Segmentation. Semantic segmentation is a high-level visual perception task that requires per-pixel high-level understanding. We evaluate our VPD on ADE20K [58] and compare it with previous backbones and pre-training methods. We start by performing experiments on the default training schedule, where we train our model with a Semantic FPN [24] head for 80K iterations. The results can be found in Table 1. For fair comparisons, we do not consider complex segmentation heads such as MaskFormer [11]. Instead, we compare the available result with more common segmentation heads like Semantic FPN [24] and UperNet [52]. The compared methods include self-supervised pre-training (MAE [17]) and supervised pre-training (Swin [28] and ConvNeXt [29]). We report both the single-scale and multi-scale mIoU for all the methods. We show that VPD can achieve 53.7 mIoU^{ss} and 54.6 mIoU^{ms}, outperforming pre-trained ConvNeXt-XL [29] model with comparable computational complexity. Notably, while other methods utilize UPerNet [52] as the segmentation head and train the model for >120K iterations, our model trained for only 80K iterations can achieve better results with a more lightweight Semantic FPN [24] head and 512×512 crop size. We further perform experiments with a faster schedule, where we train our models for only 8K iterations. We report both the single-scale and multi-scale mIoU at 4K/8K iterations, as shown in Table 2. We use A32 and A64 subscripts to represent cross-attention maps with spatial sizes up to 32 and 64, respectively. For 8K iterations, we find VPD$_{\text{A32}}$ surpass all the baseline methods, including those pre-trained on mask image modelling [17, 38], contrastive learning [7] and supervised learning [27, 29]. For 4K iterations, we show VPD$_{\text{A64}}$ can yield better results. The results indicate that VPD has the potential to enhance adaptation to downstream tasks and that incorporating additional semantic guidance from cross-attention maps can expedite its convergence even further.

Table 2. **Semantic segmentation with fewer training iterations.** We compare the performance of our VPD with previous models with different architectures and different pre-training methods. The performance is measured by the mIoU of single-scale and multi-scale at 4K/8K iterations.

Method	4K Iters		8K Iters	
	mIoU^{ss}	mIoU^{ms}	mIoU^{ss}	mIoU^{ms}
DINO-ViT-B/8 [7]	32.4	31.1	40.8	39.9
MAE-ViT-L/16 [17]	37.8	36.3	46.7	46.4
BeiTv2-ViT-L/16 [38]	32.1	33.6	42.9	44.7
SwinV2-L [27]	40.6	41.1	47.5	48.2
ConvNeXt-XL [29]	43.2	43.7	47.1	47.8
VPD$_{\text{A32}}$	43.1	44.2	**48.7**	**49.5**
VPD$_{\text{A64}}$	**43.9**	**44.7**	47.7	49.1

Referring Image Segmentation. Referring image segmentation also involves high-level knowledge of the correspondence between visual content and referring expression texts. We evaluate our VPD on the widely used RefCOCO [56], RefCOCO+ [56], and G-Ref [33]. We train our model on the training set and report the overall IoU (oIoU) on the validation set, as shown in Table 3. Under the default training schedule, our VPD outperforms previous methods by large margins consistently on both two datasets. We also find that when trained for only 1 epoch, our VPD also achieves better overall IoU than previous state-of-the-art LAVT [54]. We hypothesize that VPD achieves superior performance due to two primary reasons: (1) unlike prior methods that rely on pre-trained language models that lack interactions with the visual modality, our VPD model leverages a pre-trained

Table 3. **Referring image segmentation on RefCOCO.** We compare our VPD with previous methods with both the default training schedule and fast schedule (1 epoch) on three benchmark datasets of RefCOCO (RefCOCO, RefCOCO+, and G-Ref). The performance is measured by the overall IoU on the validation set. We show our VPD achieves better performance consistently on all three benchmarks.

Method	Language	RefCOCO	RefCOCO+	G-Ref
default schedule				
MAttNet [55]	Bi-LSTM	56.51	46.67	47.64
MCN [32]	Bi-LSTM	62.44	50.62	49.22
CGAN [31]	Bi-GRU	64.86	51.03	51.01
LTS [21]	Bi-GRU	65.43	54.21	54.40
VLT [13]	Bi-GRU	65.65	55.50	52.99
LAVT [54]	BERT	72.73	62.14	61.24
VPD	CLIP	**73.25**	**62.69**	**61.96**
fast schedule, 1 epoch				
LAVT [54]	BERT	52.56	29.17	40.31
VPD	CLIP	**63.04**	**40.01**	**48.11**

Table 4. **Depth estimation on NYUv2 [47].** We report the commonly used metrics for depth estimation including RMSE, δ_n, REL and $\log_{10}$ (see Section 4.1 for details). We show that VPD outperforms previous state-of-the-art methods consistently in all the metrics. We also demonstrate our model converges faster than SwinV2 [53] pre-trained with masked image modeling in the fast training schedule.

Method	RMSE$\downarrow$	$\delta_1 \uparrow$	$\delta_2 \uparrow$	$\delta_3 \uparrow$	REL $\downarrow$	log10 $\downarrow$
default schedule						
BTS [25]	0.392	0.885	0.978	0.995	0.110	0.047
AdaBins [4]	0.364	0.903	0.984	0.997	0.103	0.044
DPT [41]	0.357	0.904	0.988	0.998	0.110	0.045
P3Depth [37]	0.356	0.898	0.981	0.996	0.104	0.043
NeWCRFs [57]	0.334	0.922	0.992	0.998	0.095	0.041
SwinV2-B [27]	0.303	0.938	0.992	0.998	0.086	0.037
SwinV2-L [27]	0.287	0.949	0.994	0.999	0.083	0.035
AiT [35]	0.275	0.954	0.994	0.999	0.076	0.033
ZoeDepth [5]	0.270	0.955	0.995	0.999	0.075	0.032
VPD	**0.254**	**0.964**	**0.995**	**0.999**	**0.069**	**0.030**
fast schedule, 1 epoch						
SwinV2-B [27]	0.462	0.819	0.975	0.995	0.133	0.059
SwinV2-L [27]	0.381	0.886	0.984	0.997	0.112	0.051
VPD	**0.349**	**0.909**	**0.989**	**0.998**	**0.098**	**0.043**

diffusion model that learned to generate images guided by the text, thereby establishing a natural connection between language and visual domains. (2) the explicit guidance provided by cross-attention maps offers the model an effective starting point for generating accurate segmentation results.

Depth Estimation. We start by evaluating VPD on depth estimation, a visual perception task that requires low-level per-pixel understanding. We use the popular benchmark NYUv2 and compare VPD with previous methods, as shown in Table 4. Under the default training schedule, our VPD achieves 0.254 RMSE, establishing the new state-of-the-art. Notably, our method outperforms SwinV2-B/SwinV2-L [53], which uses a very strong visual backbone SwinV2 [27] pre-trained on masked image modeling. Additionally, we verify the fast convergence of VPD by training the model for only one epoch. Table 4 shows that VPD converged much faster than SwinV2-L [53]: VPD achieves 0.349 RMSE (lower is better) while the RMSE of SwinV2-L [53] is 0.381. These results further demonstrate that large-scale text-to-image pre-training can be very competitive in downstream visual perception tasks, even compared with the dedicated visual pre-training methods.

4.3. Analysis

In this section, we will conduct detailed analyses to further evaluate the effectiveness of each of the components in VPD, as well as demonstrate the scaling potential of it.

Effectiveness of components of VPD. We first evaluate the effectiveness of the components presented in Section 3, as is shown in table 5. We perform the ablation studies on semantic segmentation, using the same training configurations as Table 2. We start from a vanilla usage of the pre-trained ϵ_θ network as our baseline and add the proposed components gradually to verify the contribution of each. For our baseline (the first row), we feed an empty string as the text prompt, such that no effective visual-language interactions are introduced. We find the performance of the baseline is far from satisfactory (*e.g.*, only 46.9 mIoU at 8K iterations). We then apply the text prompts constructed by filling the class names of ADE20K [58] to the template "`a photo of a [CLS]`", which can improve the mIoU@4K and mIoU@8K by 0.5 and 0.2, respectively. This reveals that a proper text prompt can build the connection between visual and language domains. To further mitigate the domain gap, we employ the text adapter after the CLIP text encoder, which brings significant improvement (42.0→42.9 in mIoU@4K and 47.1→48.0 in mIoU@8K). Finally, we add the cross-attention maps as explicit semantic guidance (the last row of Table 5) and find the mIoU@8K can be further improved by 0.7. These ablation studies clearly demonstrate that our designs in VPD can effectively leverage the pre-trained knowledge of the ϵ_θ via both implicit and explicit guidance.

Choice of the cross-attention maps. There are a lot of cross-attention layers in the denoising autoencoder ϵ_θ. Therefore, it becomes a question that which cross-attention maps we should select to provide semantic guidance. Since ϵ_θ is implemented as a UNet [44], it consists of mainly three groups of blocks including the downsampling blocks, the middle blocks, and the upsampling blocks. Specifically, for an input image of 512×512, the corresponding size of the latent features is 64×64. The downsampling blocks first gradually reduce the spatial size of the feature maps from 64×64 to 8×8, and then feed them to the middle

Table 5. **Ablation studies.** We perform ablations in semantic segmentation on ADE20K [58] to verify the effectiveness of each of the proposed components in VPD and the influence of the different choices of cross-attention maps. We find that all the proposed components are beneficial and that combining the cross-attention maps in the downsampling blocks and the upsampling blocks yields the best performance.

text prompt	text adapter	cross attn	mIoU 4K	mIoU 8K
✗	✗	✗	41.5	46.9
✓	✗	✗	42.0	47.1
✓	✓	✗	42.9	48.0
✓	✓	mid	43.0	47.8
✓	✓	down	43.2	48.2
✓	✓	up	43.2	48.5
✓	✓	up+down	43.1	**48.7**

blocks which do not change the spatial size. Finally, the upsampling blocks progressively increase the feature map size back to 64 × 64 and merge the information via some lateral connections from the downsampling blocks.

The comparisons of leveraging the cross-attention maps from different locations can be found in the bottom part of Table 5. First, we show that using the cross-attention maps from middle blocks might be harmful to the performance, mainly because the spatial resolution is too low to provide accurate information. Second, we find that the cross-attention maps from both the upsampling blocks and the downsampling blocks can bring considerable improvements and the upsampling blocks seem to be more beneficial to the performance. This is also reasonable because the cross-attention map will become more and more accurate during the forward procedure. Finally, we average both the cross-attention maps from the upsampling and downsampling blocks and demonstrate that they cooperate well and achieve better results in both mIoU@4K and mIoU@8K.

Effects of different pre-trained weights. Since our VPD is built on pre-trained text-to-image diffusion models, it is necessary to investigate how the pre-trained weights would affect the performance of our VPD. In our previous experiments, we have used the released `1-5` version of the "Stable-Diffusion" (`SD-1-5` for short). Now we compare different releases of "Stable-Diffusion" by applying the weights in the semantic segmentation on ADE20K [58], and the results are illustrated in Figure 3. The differences between the checkpoints are the pre-training iterations on 512×512 resolution. We omit the `SD-1-3` since it is trained for only 30K fewer iterations than `SD-1-4`. Our results in Figure 3 demonstrate a clear trend that more pre-trained iterations of the text-to-image diffusion model will also exhibit better performance on downstream tasks with VPD. It is worth noting that from `SD-1-1` to `SD-1-5`, the mIoU@8K is improved by more than 4, which is quite considerable. We hypothesize that this is mainly because longer training can improve the alignment between visual and language, which can be also verified from the CLIP score of the different versions reported by the "Stable-Diffusion"[1]. These results also show that the success of our method is based on the learned visual-language knowledge rather than the large capacity of the ϵ_θ network. The upward trend in the graph demonstrates the scaling ability of our VPD, indicating that a stronger pre-trained text-to-image diffusion model can help us achieve better results.

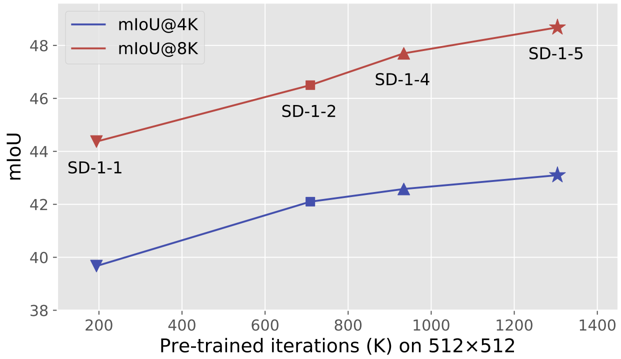


Figure 3. **Longer pre-training yields better performance on downstream tasks.** We train VPD with different versions of Stable-Diffusion (indicated by `SD-1-x`) on ADE20K and investigate how the pre-training iteration would affect the performance. The upward trend demonstrates that our VPD can benefit from a stronger text-to-image diffusion model.

Limitations. While our method has shown satisfactory performance, we acknowledge that the computational cost of VPD is currently relatively high. Unlike recognition models that are explicitly designed to balance efficiency and accuracy, generative models prioritize synthesis quality and often lack careful consideration of complexity. Although we have demonstrated the potential of extracting valuable information from a pre-trained text-to-image diffusion model, the high computational costs of ϵ_θ cannot be addressed within our current framework. We believe that further improvements in the complexity-accuracy trade-offs of VPD can be achieved through a more lightweight design of the generative model or a more efficient architecture dedicated to both generative and perception tasks.

5. Conclusion

In this paper, we have proposed a new framework called VPD to transfer the high-level knowledge of a pre-trained text-to-image diffusion model to downstream tasks. We have proposed several designs to encourage visual-language alignment and prompt the pre-trained model implicitly and explicitly. Extensive experiments on semantic segmentation, referring image segmentation, and depth estimation have demonstrated that VPD can achieve very competitive

[1] see https://github.com/runwayml/stable-diffusion for details.

performance and exhibits faster convergence compared to methods with various visual pre-training paradigms. We also believe that text-guided generative models other than diffusion models[45, 40, 8] can also fit in VPD, which we leave to future work. We expect our efforts to shed light on the crucial role of generative text-to-image pre-training in visual perception and make a step towards the unification of visual generation and perception tasks.

References

[1] Tomer Amit, Eliya Nachmani, Tal Shaharbany, and Lior Wolf. Segdiff: Image segmentation with diffusion probabilistic models. *arXiv preprint arXiv:2112.00390*, 2021. 3

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

[3] Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schönlieb, and Christian Etmann. Conditional image generation with score-based diffusion models. *arXiv preprint arXiv:2111.13606*, 2021. 4

[4] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *CVPR*, pages 4009–4018, 2021. 7

[5] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023. 7

[6] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022. 1

[7] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, pages 9650–9660, 2021. 3, 6

[8] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023. 9

[9] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. Diffusiondet: Diffusion model for object detection. *arXiv preprint arXiv:2211.09788*, 2022. 3

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pages 1597–1607. PMLR, 2020. 2

[11] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *NeurIPS*, 34:17864–17875, 2021. 6

[12] 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*, pages 248–255. Ieee, 2009. 3

[13] Henghui Ding, Chang Liu, Suchen Wang, and Xudong Jiang. Vision-language transformer and query generation for referring segmentation. In *ICCV*, pages 16321–16330, 2021. 7

[14] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *NeurIPS*, 27, 2014. 5

[15] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021. 4

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

[17] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, pages 16000–16009, 2022. 2, 3, 6

[18] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *CVPR*, 2020. 2, 3

[19] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 1, 3, 5

[20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 33:6840–6851, 2020. 2, 3

[21] Ya Jing, Tao Kong, Wei Wang, Liang Wang, Lei Li, and Tieniu Tan. Locate then segment: A strong pipeline for referring image segmentation. In *CVPR*, pages 9858–9867, 2021. 7

[22] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *NeurIPS*, 34:21696–21707, 2021. 3

[23] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2

[24] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *CVPR*, pages 6399–6408, 2019. 2, 4, 5, 6

[25] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *arXiv preprint arXiv:1907.10326*, 2019. 7

[26] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *ICLR*, 2022. 2

[27] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *CVPR*, pages 12009–12019, 2022. 6, 7

[28] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, pages 10012–10022, 2021. 6

[29] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *CVPR*, 2022. 2, 6

[30] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *NeurIPS*, 2022. 2

[31] Gen Luo, Yiyi Zhou, Rongrong Ji, Xiaoshuai Sun, Jinsong Su, Chia-Wen Lin, and Qi Tian. Cascade grouped attention network for referring expression segmentation. In *ACM MM*, pages 1274–1282, 2020. 7

[32] Gen Luo, Yiyi Zhou, Xiaoshuai Sun, Liujuan Cao, Chenglin Wu, Cheng Deng, and Rongrong Ji. Multi-task collaborative network for joint referring expression comprehension and segmentation. In *CVPR*, pages 10034–10043, 2020. 7

[33] Varun K Nagaraja, Vlad I Morariu, and Larry S Davis. Modeling context between objects for referring expression understanding. In *ECCV*, pages 792–807. Springer, 2016. 2, 6

[34] Varun K Nagaraja, Vlad I Morariu, and Larry S Davis. Modeling context between objects for referring expression understanding. In *ECCV*, pages 792–807. Springer, 2016. 5

[35] Jia Ning, Chen Li, Zheng Zhang, Zigang Geng, Qi Dai, Kun He, and Han Hu. All in tokens: Unifying output space of visual tasks via soft token. *arXiv preprint arXiv:2301.02229*, 2023. 7

[36] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. *arXiv preprint arXiv:2302.03027*, 2023. 1

[37] Vaishakh Patil, Christos Sakaridis, Alexander Liniger, and Luc Van Gool. P3depth: Monocular depth estimation with a piecewise planarity prior. In *CVPR*, pages 1610–1621, 2022. 7

[38] Zhiliang Peng, Li Dong, Hangbo Bao, Qixiang Ye, and Furu Wei. Beit v2: Masked image modeling with vector-quantized visual tokenizers. *arXiv preprint arXiv:2208.06366*, 2022. 3, 6

[39] 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. In *ICML*, pages 8748–8763. PMLR, 2021. 4

[40] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 1, 9

[41] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *ICCV*, pages 12179–12188, 2021. 7

[42] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *CVPR*, 2022. 6

[43] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 1, 2, 4

[44] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241. Springer, 2015. 2, 3, 7

[45] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 9

[46] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *arXiv preprint arXiv:2210.08402*, 2022. 1, 2, 4

[47] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. *ECCV*, 7576:746–760, 2012. 2, 5, 7

[48] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, pages 2256–2265. PMLR, 2015. 2, 3

[49] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv:2010.02502*, October 2020. 2

[50] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021. 4

[51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 30, 2017. 2

[52] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *ECCV*, pages 418–434, 2018. 6

[53] Zhenda Xie, Zigang Geng, Jingcheng Hu, Zheng Zhang, Han Hu, and Yue Cao. Revealing the dark secrets of masked image modeling. *arXiv preprint arXiv:2205.13543*, 2022. 6, 7

[54] Zhao Yang, Jiaqi Wang, Yansong Tang, Kai Chen, Hengshuang Zhao, and Philip HS Torr. Lavt: Language-aware vision transformer for referring image segmentation. In *CVPR*, pages 18155–18165, 2022. 5, 6, 7

[55] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. Mattnet: Modular attention network for referring expression comprehension. In *CVPR*, pages 1307–1315, 2018. 7

[56] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *ECCV*, pages 69–85. Springer, 2016. 2, 5, 6

[57] Weihao Yuan, Xiaodong Gu, Zuozhuo Dai, Siyu Zhu, and Ping Tan. New crfs: Neural window fully-connected crfs for monocular depth estimation. *arXiv preprint arXiv:2203.01502*, 2022. 7

[58] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017. 2, 5, 6, 7, 8

[59] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 130(9):2337–2348, 2022. 4