Composing Parts for Expressive Object Generation

Harsh Rangwani[1] Aishwarya Agarwal[1] Kuldeep Kulkarni[1]
R. Venkatesh Babu[2] Srikrishna Karanam[1]

[1]Adobe Research [2] Indian Institute of Science

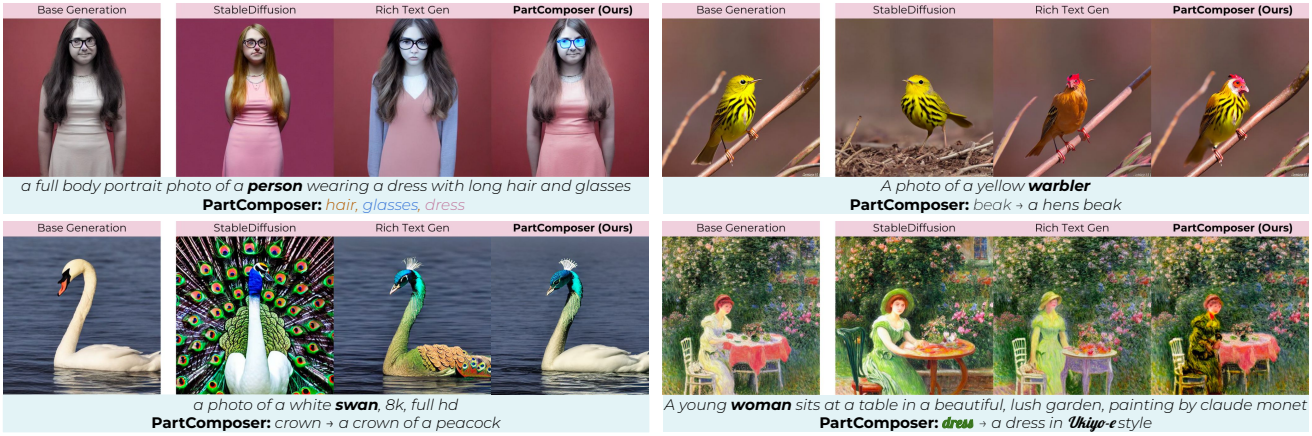


Figure 1. **Base Generation (left) Comparison with Methods for Generations with Parts Details (right).** PartComposer allows the generation of object images with specified attributes (color, style *etc*.) of parts for the **chosen object** in the base text prompt. StableDiffusion and Rich-Text [18] methods with part details either ignore the part instructions or generate inconsistent ojects.

Abstract

Image composition and generation are processes where the artists need control over various parts of the generated images. However, the current state-of-the-art generation models, like Stable Diffusion, cannot handle fine-grained part-level attributes in the text prompts. Specifically, when additional attribute details are added to the base text prompt, these text-to-image models either generate an image vastly different from the image generated from the base prompt or ignore the attribute details. To mitigate these issues, we introduce PartComposer, a training-free method that enables image generation based on fine-grained part-level attributes specified for objects in the base text prompt. This allows more control for artists and enables novel object compositions by combining distinctive object parts. PartComposer first localizes object parts by denoising the object region from a specific diffusion process. This enables each part token to be localized to the right region. After obtaining part masks, we run a localized diffusion process in each part region based on fine-grained part attributes and combine them to produce the final image. All stages of PartComposer are based on repurposing a pre-trained diffusion model, which enables it to generalize across domains. We demonstrate the effectiveness of part-level control provided by PartComposer through qualitative visual examples and quantitative comparisons with contemporary baselines.

1. Introduction

Image generation with large generative diffusion models like StableDiffusion [40], DALLE [38], etc., has become prevalent due to their superior quality and extensive world knowledge. These models are trained on large image-caption datasets and are trained to generate images based on a given text prompt (description). As image composition and creation are creative processes where the artists need control over various parts of the image being generated. However, adding additional details for controlling part appearance in the text prompt either changes the generated image entirely or ignores the part instructions [9] (Fig. 1).

Various works aim to provide improved spatial controls to image generation, as they allow image generation conditioned on segmentation masks [5], edge maps [55], bounding boxes [10, 27] etc. Popular methods such as Control-

Net [55], GLIGEN [27], etc. require specifying training of these conditional modules, which allows for the controllability of these large generative models. Further, there has been some development of training-free approaches [9, 10] which enable controlled generation of objects by modulating the internal cross-attention activations in the diffusion process. This demonstrates that these pre-trained generative models contain information about the spatial parts of the image, and modulating them effectively can lead to image compositions.

However, despite several attempts [6, 55], the semantic controllability of the image generation is still restricted to specifying details at the object level. The object level details can be restrictive, as often creative designers synthesize object parts (*e.g*. shirt, trouser etc.) and then compose them [23]. Further, variations in semantic parts are often very distinctive and are uniquely used to identify objects [12, 30]. For example, we usually identify species of birds by their unique *beak*. Due to this, semantic understanding and recognition of parts have been widely studied as a topic in computer vision [12, 13, 21, 31]. Hence, providing controllability for image synthesis at the object semantic part level can enable a large variety of image compositions [23]. Towards this goal, we introduce **PartComposer**, which provides users with an interface through which they can select the object in the scene and provide a semantic part-level description with fine-grained details for the object generation.

In **PartComposer**, we develop a scheme to extract part-level localization masks from the Diffusion model. We introduce a parallel part diffusion process that generates masks for the object parts. The core idea of the approach is *that by forcing the part diffusion model to specifically denoise only the object region in the image, it is possible to understand the locations of various parts in the object*. After denoising just the object through the part model, the information inside the part diffusion model present in attention maps can be used to generate the masks for various object parts. In the following Part Generation stage, PartComposer utilizes the part masks and attributes of each part the user provides. We enable users to provide a highly expressive specification of parts by using a Rich-Text [18] interface, which allows the specifying attributes, such as style, color, etc., for each part. For the final image generation, taking inspiration from recent studies like Rich-Text Generation [18], Multi-Diffusion [5] etc., we compose the various object parts, by running parallel masked diffusion process for each part while combining them periodically into the image. This combination enables harmonious composition of parts, and also masking ensures only local modifications to the regions corresponding to each part of the object specified. In the PartComposer method, we only use the pre-trained StableDiffusion model, making it a generalizable and training-free approach.

We extensively test the proposed PartComposer approach for the zero-shot object part segmentation, which is a challenging setup in computer vision. We evaluate the part segmentation approach on DeepFashion [29] and CUB200 [50] datasets, where our method significantly outperforms the baseline StableDiffusion (SD). Further, to evaluate the PartComposer image composition abilities, we also provide quantitative results along with a user preference study, where our method significantly outperforms the baselines in generating images consistent with the described parts (Fig. 1). We summarize the core contributions of our paper below:

1. We introduce PartComposer, a training-free method that enables the generation of object images by using provided fine-grained details for the parts of the object. For example, while generating a bird image we can specify a detailed description of its beak.
2. In PartComposer, we introduce a novel Part-Diffusion process, which localizes and provides masks for parts of a base object generated by the Diffusion model (Fig. 3). To localize object parts, we introduce a novel segmentation scheme that uses the attention maps of the base diffusion and part diffusion process to obtain accurate masks for localized parts.
3. In PartComposer, after localization, we enable the generation of parts from the pre-trained diffusion model based on the Rich-Text description for the parts provided by the user. PartComposer then composes the image to harmoniously blend all object parts with the background by combining localized diffusion paths (Fig. 2).

2. Related Works

Text-to-Image Generative Models. The text-to-image models synthesize images by following a textual description provided as a prompt. These models have recently become mainstream due to their superior image generation quality and significant knowledge base. This is an outcome of the availability of large-scale image caption datasets [42] and highly parallelized GPU clusters. Almost all kinds of generative models, such as GANs [22], Autoregressive [41, 53], and Diffusion models [38, 40], have shown significant improvements in quality with training on these large image caption datasets. Among these models, the StableDiffusion (SD) [40] models, based on denoising diffusion in latent space are popular due to their open-source nature, which we also utilize for experiments in our work.

Text-to-Image Models for Downstream Tasks. Generative models, in general, have been useful for various downstream tasks, particularly ones based on per-pixel prediction like Segmentation [1, 11], Depth Estimation [7], etc. As layers near the image generation output have features that capture the pixel-pixel relation [7]. With the large-scale text-to-image generative models, these models often perform very competitively [46, 52] to discriminative methods, on tasks like segmentation. However, one commonality among most of these segmentation methods is that they operate at the

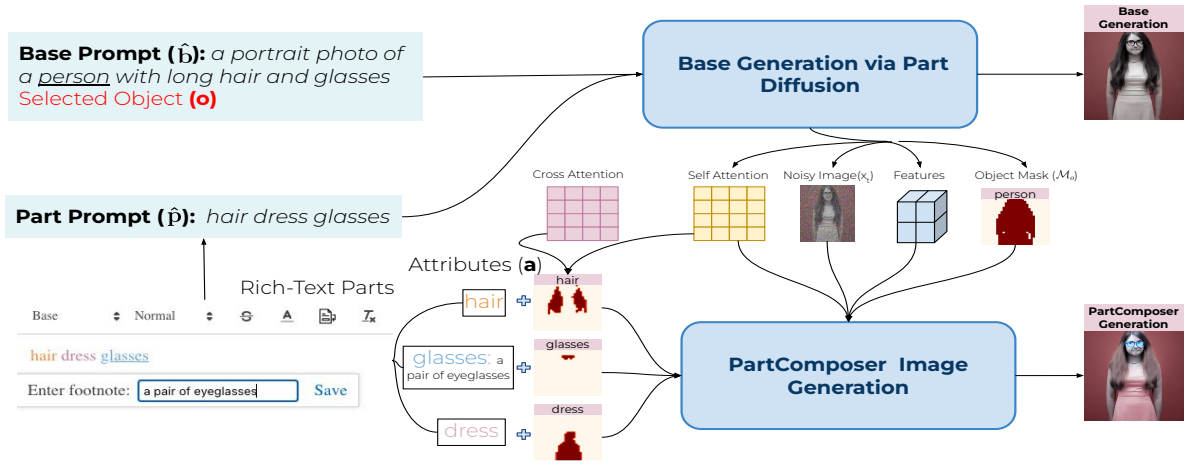


Figure 2. **PartComposer** takes the input of a base prompt, the selected object **o**, and a Rich-Text description (i.e., footnote, colors, etc.) of the parts. The Part Diffusion process generates the object masks for the specified parts. Then, PartComposer runs a parallel Region Diffusion process to generate attributes of the specified parts, guided by base generation's intermediate outputs.

granularity of object or instance level. Text-to-image models here are at an advantage as the usual image captions describe the scene at an object level. In this work, we take a step further in exploring the object part-level knowledge of these text-to-image diffusion models.

Part Discovery and Segmentation. Part Discovery and Attribution were an integral part of computer vision pipelines classically, as these approaches were robust to viewpoint variations [24, 31, 48]. In deep learning, the unsupervised (self-supervised) approaches for part discovery like SCOPS [21] and Unsup-Parts [12] became popular as they generalize to object parts across categories. In this work, we go one step ahead and operate in a zero-shot unsupervised part segmentation setting, generating part masks from T2I models.

Controllable Image Generation. Masks, bounding boxes, edge maps, depth maps, etc., have been explored to control the generations of text-to-image diffusion models [6, 14, 27, 35, 36, 55] in addition to text. Further, various other approaches [2–4, 9, 34, 45, 49, 51] achieve control of image semantics through modulating diffusion models. Despite this, the text-to-image generation control at the object part level is under-explored; a recent work [33] tries to do it in a controlled supervised setup using part-masks. Contrary to that, in our work PartComposer we explore a generalized training-free zero-shot setting.

3. Method

In this section, we introduce PartComposer, our method to synthesize objects based on the description of parts of the objects. In PartComposer, we ask the user to specify a base prompt and the token for the object for which it wants to synthesize parts. Then, we provide the user a *rich-text* [18] editor (Fig. 2) to specify the parts and their description. PartComposer involves two diffusion steps, **a) Part Localization:** In the early diffusion stage, we get a mask for the object we want to divide into parts. Then, we perform denoising in later stage from a U-Net condition on parts to fill the masked region of the object, during which it learns to denoise different object parts. Due to this, the attention maps for various parts highlight the correct part region, which we use to extract the part mask. The infilling process is the major contribution of the PartComposer method (Fig, 3). **b) Part Generation:** For generating parts, we combine region-specific diffusion processes for various parts by iteratively merging them inspired by MuliDiffusion [5], Rich-Text Generation [18] etc. However, till now, most of these works have combined the diffusion process for generating objects; for the first time, we have demonstrated its effectiveness in generating object parts. We provide an overview of the PartComposer pipeline in Fig. 2.

3.1. Problem Setup

We consider each span of tokens $\mathbf{p}_i$ as indicative of describing one part of the object, with attributes $\mathbf{a}_i$ describing its overall appearance. Our design choices are based on the rich-text generation [18], and we allow the user to specify the following type of attributes for part generation (Fig. 2):

Part Description (*i.e.* footnote). It is an important attribute that specifies the part of an object; for example, for the part ($\mathbf{p}_i$) 'crown' of a bird, we can specify the attribute ($\mathbf{a}_i^p$ = 'a crown of a peacock'). This helps specify novel part descriptions, which can lead to artistic novel compositions.

Font Color. It helps us specify the exact RGB values for the color attribute $\mathbf{a}_i^c$ we want for the object part $\mathbf{p}_i$. The exact value of RGB allows fine-grained control over the color of the desired part, whereby just specifying specific colors like 'brick red' leads to ignorance by Stable Diffusion [18].

Font Style. The font style is indicative of the specific artistic style $\mathbf{a}_i^s$ like 'of Claude Monet' and 'of Van Gogh' when

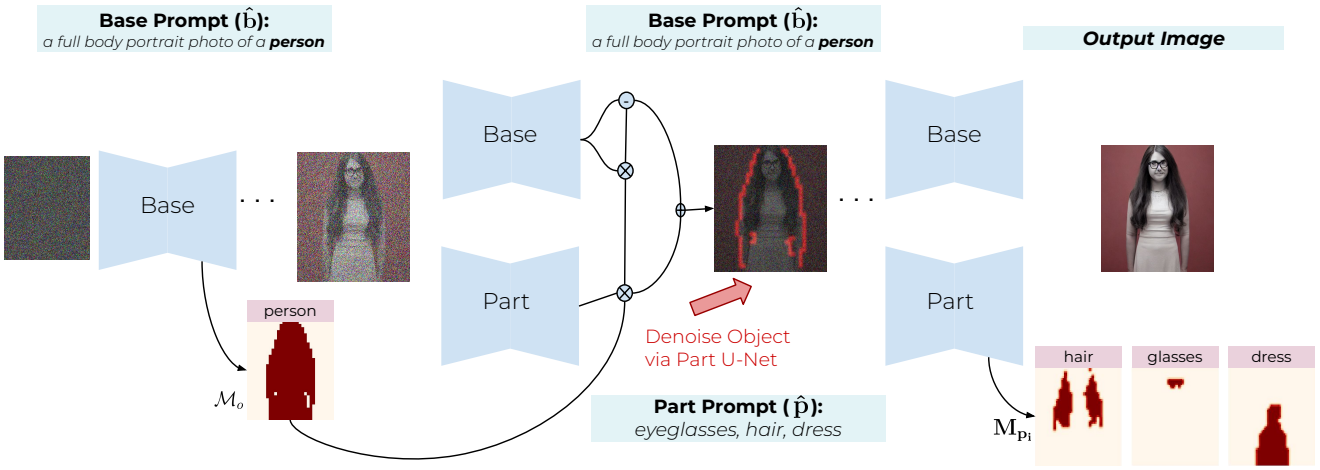


Figure 3. **Localization.** We obtain the mask ($\mathcal{M}_o$) for object in middle of diffusion process. We then denoise object in masked region, using parts $\mathbf{p}_i$ conditioned U-Net. Due to part denoising, the attention maps of $\mathbf{p}_i$ can now localize parts as masks $\mathbf{M}_{\mathbf{p}_i}$.

synthesizing images of paintings. This instructs the model to generate a part following a specific style for the given prompt and blend it with other parts of the object (Sec. 5).

Font Size. The font size controls the relative size of each part in a generation [19]. We use the $\mathbf{a}_i^w$ to denote size.

3.2. Part Localization

For part localization, we first run the base stable diffusion model for the given text prompt and extract the token map $\mathcal{M}_o$ for the object $\mathbf{o}$ specified by the user, by following the technique of clustering the self-attention maps [18, 37]. We obtain the mask using this clustering after denoising between the initial step (T) until a threshold time step T_{th}. After obtaining the binary object mask $\mathcal{M}_o$ using attention masks [18, 37], we run two parallel diffusion processes where one contains input from the base prompt and the other contains input from the part prompts $\hat{\mathbf{p}} = [\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_n]$, a token $\mathbf{p}_i$ for each of the parts specified in the part text prompt. We denote the $\hat{\mathbf{b}}$ to denote the base text prompt and $\hat{\mathbf{p}}$ to denote the part prompt. For the time $t \leq T_{th}$ (by default, we use $T_{th} \approx T/2$), we denoise the U-Net in the object region by denoising it with both the combination of the Part Prompt output and Base Prompt output. Due to this, the Part-Based U-Net gets the information regarding various parts in the object (Fig. 3). We now mathematically define the output noise ϵ_t for the diffusion process with Part-Diffusion below:

$$\epsilon_t = \alpha\mathcal{M}_o \odot D(x_t, \hat{\mathbf{p}}, t) + (1 - \alpha\mathcal{M}_o) \odot D(x_t, \hat{\mathbf{b}}, t) \quad (1)$$

here, the α is the hyper-parameter controlling the strength of the part prompt diffusion output, and D is the output of the pre-trained U-Net of a text-to-image diffusion model. The above denoising process is followed for t steps until the last step to produce a base image corresponding to the given prompt $\hat{b}$. Keeping a high T_{th} and low α, makes minimal changes in output as if the denoising diffusion was done with original prompt $\hat{b}$ (See Suppl. Sec.). With this part denoising of object, we obtain attention masks, from which we extract the localization information of part $\mathbf{p}_t$. We now describe the process of obtaining the part masks.

Token Maps for Parts. We first take the part tokens $\hat{\mathbf{p}}$, which are a concatenation of the part names (*i.e.* 'beak crown wings'), which may not make a meaningful text prompt. Hence, we initialize text embeddings for all these tokens $\mathbf{p}_i$ by passing the meaningful text prompt having the following template: " A photo of $\mathbf{p}_i$ of a $\mathbf{o}$ " where $\mathbf{p}_i$ is the object part name and $\mathbf{o}$ is the object name. This serves two purposes: first, it makes the text embedding meaningful, and second, it introduces some invariance from the order of part specification in part prompt. These embeddings are then passed as text embeddings to the Part U-Net for denoising. After running the denoising process, we aggregate the self-attention maps across multi-heads and time steps (from 32 $\times$ 32 resolution) for both the base and part U-Net diffusion branches, taking inspiration from works that demonstrate that attention can localize objects [9, 18, 30, 44]. We then perform spectral clustering on these attention maps to form k segmentation maps $\hat{\mathbf{M}}$ (32 $\times$ 32), based on pixel-pixel similarity. To attribute these K-segments to the part specified by the user, we aggregate the cross attention of Part U-Net diffusion process. For each token $\mathbf{p}_j$, we obtain the cross-attention score as follows: $\hat{\mathbf{m}}_j = \frac{c_j}{\sum_k c_k}$, where c_j is the cross attention score for each token. We proceed by aggregating the attention heads to obtain the average cross-attention scores and resizing them to 32 $\times$ 32, obtaining $\hat{\mathbf{m}}$. We remove the start of text ($\langle$sot$\rangle$) token for cross-attention and re-normalize it [9]. In other works [18, 37], as the tokens correspond to the objects being generated in the image, it's sure that token maps will be meaningful. However, this is not true for parts, as Part U-Net might not localize some

parts. For determining if the part is localized, *we propose to look at the max value of cross attention map spatially across the pixels*; if we find that the following condition is met, the part is localized:

$$L(j) = \mathbb{1}\{\max(\hat{\mathbf{m}}_j) \geq (1-\delta)\frac{1}{K}\}. \quad (2)$$

Here δ is a hyperparameter, and K is the number of parts. is This condition is robust in finding the localized parts (See Sec. 5 for ablation). For the parts $\mathbf{p}_i$, which are localized, we normalize the cross-attention map:

$$\hat{\mathbf{m}}_j = L(j)\frac{\hat{\mathbf{m}}_j - \min(\hat{\mathbf{m}}_j)}{\max(\hat{\mathbf{m}}_j) - \min(\hat{\mathbf{m}}_j)} + (1 - L(j))\ \hat{\mathbf{m}}_j.$$

We follow a dot-product-based protocol to assign each K cluster in the self-attention masks to a part, unlike the average attention protocol in the previous works [18, 37]. We find that dot product of normalized cross attention scores $\hat{\mathbf{m}}_j$ for each token with self-attention masks $\hat{\mathbf{M}}_j$ works better, as the attention maps in Part U-Net are noisy. Still, they are often correct for the regions that are localized in one specific area only (Suppl. Fig. 14). Hence, the dot product protocol favors those maps that are only localized in some areas of the image and don't have high attention values across all parts of the object. After obtaining the dot product scores for each part, we assign $\hat{\mathbf{M}}_j$ to $\mathbf{p}_i$ with the highest scores. The mask for the part $\mathbf{p}_i$ is finally given as the following:

$$\mathbf{M}_{\mathbf{p}_i} = \{\cup_j\ \hat{\mathbf{M}}_j | \operatorname{argmax}_i \hat{\mathbf{M}}_j \cdot \hat{\mathbf{m}}_i = i \text{ and } \hat{\mathbf{M}}_j \cdot \hat{\mathbf{m}}_i \geq \epsilon, \} \quad (3)$$

Here ϵ is the hyperparameter which controls the minimum similarity required between the attention mask and the part token. We combine the additional attention masks unassigned to any token, and name them as the background (other) token $\mathbf{M}_b$.

3.3. Part Generation

We follow a similar protocol for the generation of part segments as in Rich-Text Generation. We tailor the rich text generation to compose the part regions of the object in the image in place of the original scene composition. We describe the part generation protocol below briefly and refer readers to Rich-Text Gen [18] for more details. For each part $\mathbf{p}_i$ we run a region diffusion process, which runs in parallel for all the parts. We then combine the region diffusion processes to obtain the final noise prediction ϵ_t as the masked $\mathbf{M}_{\mathbf{p}_i}$ sum of the denoiser outputs:

$$\epsilon_t = \sum_i \mathbf{M}_{\mathbf{p}_i}\epsilon_{t,\mathbf{p}_i} = \sum_i \mathbf{M}_{\mathbf{p}_i} \odot D(x_t, f(\mathbf{p}_i, \mathbf{a}_i), t) \quad (4)$$

where D is the pre-trained U-Net model, and $f(\mathbf{p}_i, \mathbf{a}_i)$ is the text description of the part $\mathbf{p}_i$ constructed using the following

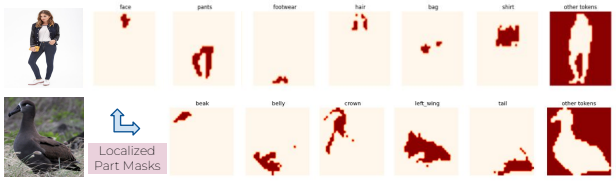


Figure 4. **Segmentation masks** for the parts that are localized by Part Diffusion for DeepFashion (above) and CUB200 (below).

process using the part tokens $\mathbf{p}_i$ and attributes $\mathbf{a}_i$. Intially the the text $f(\mathbf{p}_i, \mathbf{a}_i) = \mathbf{p}_i$, is set to part token itself. In case the part description (i.e. footnote) is available we set the $f(\mathbf{p}_i, \mathbf{a}_i) = \mathbf{a}_i^p$, further if the style attribute is available we do $f(\mathbf{p}_i, \mathbf{a}_i) = f(\mathbf{p}_i, \mathbf{a}_i) +$ in style of $+\ \mathbf{a}_i^s$. In case the color attribute is also specified, the nearest named color $\hat{\mathbf{a}}_i^c$ (*e.g.* red) for the specific RBG color $\mathbf{a}_i^c$ is found, and $f(\mathbf{p}_i, \mathbf{a}_i) = \hat{\mathbf{a}}_i^c + \text{`\ '} + f(\mathbf{p}_i, \mathbf{a}_i)$. The string $f(\mathbf{p}_i, \mathbf{a}_i)$ is the text input for the Diffusion to generate the part $\mathbf{p}_i$. We use the base prompt $\hat{\mathbf{b}}$ as $f(\mathbf{p}_i, \mathbf{a}_i)$ for the background masked region $\mathbf{M}_b$. Combining different diffusion outputs at every time t helps generate a harmonious image after blending the defined parts of the object.

Following Rich-Text [18], we also utilize the gradient guidance [16, 20] by taking the gradient of MSE loss between the estimated original image and the color value $\mathbf{a}_i^c$ specified by the user. The gradient guidance helps generate the *exact RGB color* for part $\mathbf{p}_i$, which is impossible with just text guidance [18]. Further, as we use the base text framework of Rich-Text, we also can specify font size attribute to $\mathbf{a}_i^f$ to control the relative importance of each part in the object image.

Preservation of Other Parts. As we only intend to modify the object $\mathbf{o}$ from the original prompt, we also use the Self-Injection techniques from Plug and Play [47] to maintain the overall structure of the background from base prompt generation. Further, to ensure that our diffusion trajectory follows the same path as the base, the background region is also blended with base noise generation outputs.

$$x_t = \mathbf{M}_b \odot x_t^{\text{base}} + (1 - \mathbf{M}_b) \odot x_t \quad (5)$$

In our case, the attention maps out of the object Mask $\mathcal{M}_o$ and all the object parts not assigned to any part token comprise the background $\mathbf{M}_b$, and we start this blending process at $t = T_{blend}$. We find this to be very useful in preserving the structure of the other parts of the image and just generating the described parts in the localized region. We provide an overview of the complete process in Fig. 2.

4. Experimental Analysis

4.1. Evaluation of Part Localization

We first evaluate the part localization module, which is based on the novel idea of denoising only the object region with the part-based diffusion outputs.

Table 1. **Comparison to Prior Works** for the unsupervised part segmentation task. We follow Unsup-Part [12] for evaluation protocols and baseline results for (K=4) parts. We report clustering-based NMI and ARI metrics, which are higher for better segmentation outputs.

	DeepFashion Dataset				CUB200 Dataset			
Method	FG-NMI	FG-ARI	NMI	ARI	FG-NMI	FG-ARI	NMI	ARI
				Unsupervised Learning				
SCOPS [21]	30.7	27.6	56.6	81.4	39.1	17.9	24.4	7.1
Unsup-Parts [12]	44.8	46.6	68.1	90.6	46.0	21.0	43.5	19.6
DFF [11]	-	-	-	-	32.4	14.3	25.9	12.4
				Unsupervised Zero-Shot				
Rich-Text [18]	16.0	5.2	**48.3**	58.7	3.1	0.3	3.1	0.3
StableDiffusion	12.0	3.5	40.6	70.9	8.0	0.6	3.3	0.6
PartComposer(Ours)	**24.7**	**18.0**	**48.0**	**73.4**	**20.5**	**9.2**	**18.5**	**7.7**

Implementation Details. We use the StableDiffusion (SD) version 2.1 for our experimentation purposes. We use the DDIM Scheduler [43] with 50 steps to generate results for SD2.1 to evaluate the part localization process. As ground truth is not available for the part masks of the generated images, we use test sets of the commonly used DeepFashion [29] and CUB-200 [50] datasets for evaluation. These datasets are the standard datasets used for the evaluation of the unsupervised part segmentation approaches. The DeepFashion dataset contains images along with their part segmentation masks, divided into 14 categories of labels. The CUB200 dataset contains the key point annotations for the 14 key points specified for the bird categories. We provide further details in Suppl. Section.

Baselines and Problem Setting. As we operate in the setting of Zero-shot (*i.e*. no training) Unsupervised Part Segmentation, there are no previous works that report results in such a challenging setting. Hence, we provide results for the unsupervised learning approaches to facilitate comparison. We provide results for the SCOPS [21], which utilizes the internal features of the VGG model to train a model based on self-supervised loss functions to predict parts robustly across categories. The other stronger self-supervised baseline is Unsup-Parts [12], which uses contrastive loss functions to train a network based on equivariance and other vision properties to cluster the object regions into semantic parts. In addition to this, we also report results for the DFF [11] as they also operate in the same setting. We want to highlight that these unsupervised approaches *require either training a neural network or performing clustering on the complete training data to segment parts*. In contrast, our approach is *training free and operates in a zero-shot fashion*. Hence, the performance of zero-shot approaches is not directly comparable to unsupervised methods.

Zero Shot Unsupervised Part Segmentation. As there is no benchmark to evaluate the part localization for generated images, we use the existing dataset of DeepFashion and CUB200 to obtain our results. To obtain segmentation for each image, we first invert the image into the diffusion latent space using Null-Text Inversion [32] method (see Sec. 5 for ablation). We use the BLIP-V2 [25] captioner provided in Diffusers to obtain approximate text prompts for inversion. After providing the desired image and prompt, the Null-Text inversion provides us with inverted latent and unconditioned time embeddings to reconstruct the image of the dataset. We then construct a StableDiffusion baseline in which, in addition to the prompts, we append the list of part tokens $\mathbf{p}$ to the prompt. We then use the segmentation algorithm described above in Sec. 3.2 to extract token maps corresponding to the part tokens $\mathbf{p}_i$. We also evaluate a Rich-Text [18] baseline segmentation algorithm for the same. For the proposed part-denoising approach based on parallel diffusion in PartComposer, we first get the inverted latents and embeddings. Then, we generate a base diffusion process to reconstruct the image and use Part Diffusion to fill the parts of the image. To very fairly compare the StableDiffusion performance with part-denoising, we keep all things the same except the part diffusion process to get masks. Further across these baselines, we use low classifier guidance to ensure proper reconstruction of the dataset images (See Suppl. for details).

Evaluation Protocol. We use the standard experimental protocol as used by earlier works [12] to facilitate the right point of comparison. We want to point out that in our work, the part name (*e.g*., beak, etc.) is associated with the localized mask, but the unsupervised approaches produce (K=4) parts without any part names. Hence, to make a fair comparison, we create 4 clusters of parts based on their locality, and finally generate segmentation masks with a maximum of 4 clusters. We provide the exact mapping of the part names to cluster labels in the Suppl. Section. We compare the specifically designed metrics [12] of NMI (Normalized Mutual Information) and ARI (Adjusted Rand Index) of the predicted cluster labels with part masks in the case of DeepFashion and key points in case of the CUB dataset. We report the metrics along with their foreground variants (FG-NMI and FG-ARI) for a holistic comparison of performance.

Figure 5. **Qualitative Comparison for PartComposer.** (above) We provide base prompt 'a photo of a flamingo' and part prompt as 'beak - a pelicans beak'. (below) We generate a photo of a man, with part prompt as 'black jacket and blue jeans'. We find that other baselines either ignore instruction or change the entire composition. On the contrary, PartComposer can correctly localize parts and generate details.

Results. We tabulate the results for all the baselines in Table 1. Our approach, PartComposer, significantly improves over the baselines by ≥ 5 points in NMI and ARI across most metrics in the Unsupervised Zero-Shot setting. Further, in some cases our methods, NMI and ARI, are near the unsupervised learning approaches. This demonstrates that non-trivial part masks can be generated by harnessing the power of the pre-trained diffusion models, and they can serve as a good initial prior for learning part segmentation approaches by using them as base pre-trained models. We provide qualitative results for PartComposer, where we observe that PartComposer can associate the right part with the correct label (Fig 4). On the contrary, in the StableDiffusion baseline, the parts often get assigned to the wrong part tokens, which is the cause of degraded performance (Suppl. Fig. 16). We also observe that in the case of our algorithm, if the part gets localized based on the max condition defined in the segmentation algorithm, the part mask found is usually in the right region (See Fig. 4 and Table 2). This demonstrates the effectiveness of our part segmentation procedure, which often sides with not localizing objects rather than performing arbitrary assignments.

4.2. PartComposer Image Generation Evaluation

We now evaluate the object composition ability of the proposed approach, PartComposer, compared to the baselines. In this section, we use the base generation model as a StableDiffusion (SD) 1.5 model to do a fair comparison with the Rich-Text baseline. We used the same generation setting of a PNDM [28] scheduler with 41 steps and suggested classifier guidance 8.5. To perform a fair comparison, we keep all the parameters same as that of the Rich-Text [18] baseline.

Baselines. For the task of generating the object image based on part-level details of the specified object, we use the strong baselines of Rich-Text Generation and InstructPix2Pix [8]. The Rich-Text Generation method generates the base image itself, whereas the InstructPix2Pix method requires us to generate the base image. We also evaluate the standard StableDiffusion baseline in which we add all the part details in the text prompt to generate the desired image. We compare all these baselines with our proposed method, PartComposer, while ensuring that the base parameters and seed are the same for the base image generation. A recent work [33] regarding part generation operates in a fully supervised setup using part masks and cannot be used to create novel parts based on text instruction, making it incomparable. We defer specific details for the baselines to the Suppl. Section.

Visual Comparison. We provide a visual comparison for the baselines for the a) bird image distinctive part generation and b) human image part generations. Across both cases (Fig. 5), our proposed method, PartComposer, only modifies the desired beak region and replaces it with the iconic pelican beak. The StableDiffusion baseline modifies the image completely while ignoring instructions (Supp. Fig. 12). In the Rich-Text generation, the full region corresponding to the bird gets modified instead of only the specified prompt in base generation. For the editing-based InstructPix2Pix method, we observe that it modifies the base image at a global level and cannot localize modification to the desired part region (Fig. 5). We further compare and find that recent models like SD3.5, Inpainting [17] and recent SotA editing methods [15, 26, 47, 54] can still not follow exact part instructions as seen in Supp. Fig. 8, 9 and 10. In contrast,observe that PartComposer-generated images are novel aesthetic compositions that follow the part-level instructions.

Quantitative Evaluation. As the part-specific generation is fine-grained, we perform the both automatic and user-study evaluations. We performed a user study by inviting 50 participants (28 responded with a full survey). We evaluate the model by asking questions and evaluating metrics on three orthogonal aspects i.e., a) **Localization** of part generation in comparison to the base image (through LPIPS [56]), b) **Text-Consistency** of generated parts by measuring CLIP

Table 2. **Ablation Analysis** of PartComposer for Part Localization, showing performance improvement with each component.

PartComposer (Ablations)		
Method	FG-NMI	FG-ARI
PartComposer	35.4	11.0
w/o Null-Text Inversion	23.1	5.2
w/o Max Localization	21.3	2.8
w/o Dot Product Localization	23.7	5.0
w/o Independent Text	31.2	8.9
PartComposer (Clustering)		
PartComposer (K = 9)	35.4	11.0
PartComposer (K = 4)	20.4	2.8
PartComposer (K = 14)	35.2	10.3

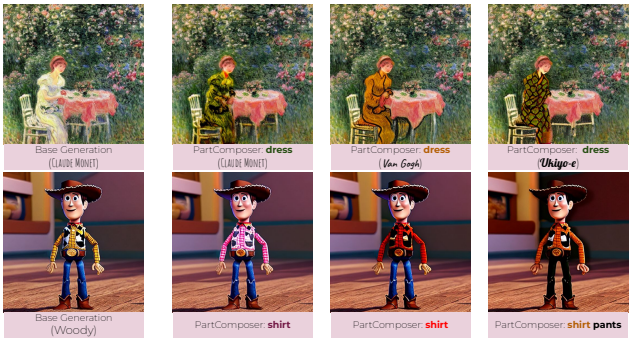


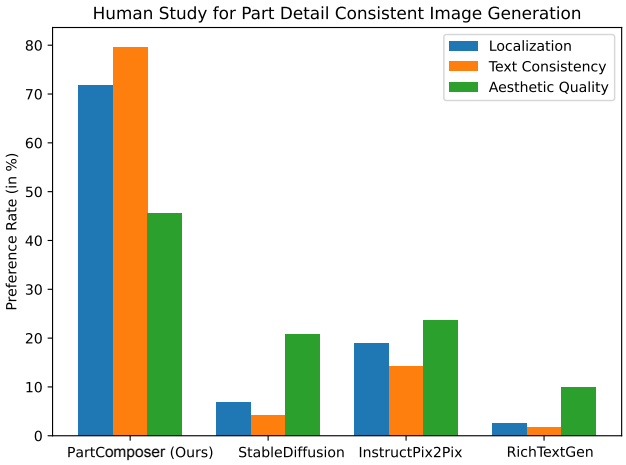
Figure 6. **PartComposer generalizes over domains** as shown by Claude Monet's painting style (above), where we specify the dress part of women to follow styles like Van Gogh and Ukiyo-e. We perform similar modifications to Woodie's image.

similarity of the part detailed text prompt with the generated image and c) **Aesthetic Quality** of the generated image through LAION-5B [42] CLIP Aesthetic scorer. In total, we collect about 3.5k opinions. In Fig. 7, we summarize the results in which we observe that PartComposer is significantly preferred over the other baselines in terms of localized and consistent Part Generation. The automated evaluation results are provided in the table below, which also follow a similar trend as in the user study, demonstrating the effectiveness of PartComposer in a localized part generation while ensuring the aesthetics score is similar to that of the base model.

5. Analysis and Discussion

Part Diffusion Based Localization. We ablate the components we have introduced in the PartDiffusion process of PartComposer (Sec. 3.2). We provide an analysis of the effect of using **a)** null-text inversion, **b)** max-based localization **c)** usage of dot product-based protocol in the part assignment, and **d)** independent text embeddings. We have used a subset of CUB-200 images to perform all evaluations, which is kept fixed across ablations. We tabulate the ablations in

Figure 7. **User Study and Quantitative Results** for part-based image generation baselines.



	PartComposer	StableDiffusion	InstructPix2Pix	Rich-Text
LPIPS	**0.168**	0.467	0.189	0.243
CLIP	**0.201**	0.183	0.193	0.187
Aesthetic	5.66	5.68	5.63	5.65

Table 2. We observe that all the components introduced in PartComposer contribute significantly to the performance.

Generalization of PartComposer. In Fig. 6, above a painting in Claude Monet style generated by the base StableDiffusion method. We then use PartComposer to specify the **dress** part of the women's token, to dark green color in Monet Style, orange color dress in Van Gogh Style (middle), and green color dress in Ukiyo-e style. In Fig. 6, we provide results for the 'Woody' character from Toy Story. These results show that PartComposer can generate shirt, pants, and dress color variations, leading to aesthetic image combinations in synthetic and natural (Supp. Fig. 13) domains . This zero-shot generalization across domains demonstrates the creative activities that can be enabled with PartComposer.

6. Conclusion

In this work we introduce PartComposer, a method to generate object images with fine-grained attribute details specified at the part level, using an expressive Rich-Text interface. The PartComposer method introduces a novel part diffusion process, which is responsible for denoising objects using the part features, and then utilizes a region-specific diffusion process to generate part details and compose the final image. PartComposer serves as initial work enabling rich-text-based training-free part-level control for SD models.

Limitations. We find that part generation is bottlenecked with the part understanding. If the part is localized correctly, the text-to-image model can generate specified details in PartComposer. Hence, improving the part-level localization of these models is a good direction for future works.

References

[1] Rameen Abdal, Peihao Zhu, Niloy J Mitra, and Peter Wonka. Labels4free: Unsupervised segmentation using stylegan. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13970–13979, 2021. 2

[2] Aishwarya Agarwal, Srikrishna Karanam, K J Joseph, Apoorv Saxena, Koustava Goswami, and Balaji Vasan Srinivasan. A-star: Test-time attention segregation and retention for text-to-image synthesis, 2023. 3

[3] Aishwarya Agarwal, Srikrishna Karanam, and Balaji Vasan Srinivasan. Training-free color-style disentanglement for constrained text-to-image synthesis. *arXiv preprint arXiv:2409.02429*, 2024.

[4] Yuval Alaluf, Daniel Garibi, Or Patashnik, Hadar Averbuch-Elor, and Daniel Cohen-Or. Cross-image attention for zero-shot appearance transfer. *arXiv preprint arXiv:2311.03335*, 2023. 3

[5] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. 2023. 1, 2, 3

[6] Shariq Farooq Bhat, Niloy J. Mitra, and Peter Wonka. Loosecontrol: Lifting controlnet for generalized depth conditioning, 2023. 2, 3

[7] Anand Bhattad, Daniel McKee, Derek Hoiem, and David Forsyth. Stylegan knows normal, depth, albedo, and more. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[8] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. 7, 6

[9] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics (TOG)*, 42(4):1–10, 2023. 1, 2, 3, 4

[10] Minghao Chen, Iro Laina, and Andrea Vedaldi. Training-free layout control with cross-attention guidance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5343–5353, 2024. 1, 2

[11] Jaehoon Choi, Taekyung Kim, and Changick Kim. Self-ensembling with gan-based data augmentation for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6830–6840, 2019. 2, 6

[12] Subhabrata Choudhury, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Unsupervised part discovery from contrastive reconstruction. *Advances in Neural Information Processing Systems*, 34:28104–28118, 2021. 2, 3, 6

[13] Edo Collins, Radhakrishna Achanta, and Sabine Susstrunk. Deep feature factorization for concept discovery. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 336–352, 2018. 2

[14] Omer Dahary, Or Patashnik, Kfir Aberman, and Daniel Cohen-Or. Be yourself: Bounded attention for multi-subject text-to-image generation. *arXiv preprint arXiv:2403.16990*, 2(5), 2024. 3

[15] Gilad Deutch, Rinon Gal, Daniel Garibi, Or Patashnik, and Daniel Cohen-Or. Turboedit: Text-based image editing using few-step diffusion models, 2024. 7, 1

[16] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 5

[17] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024. 7, 1

[18] Songwei Ge, Taesung Park, Jun-Yan Zhu, and Jia-Bin Huang. Expressive text-to-image generation with rich text. In *IEEE International Conference on Computer Vision (ICCV)*, 2023. 1, 2, 3, 4, 5, 6, 7, 8

[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. 4

[20] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 5

[21] Wei-Chih Hung, Varun Jampani, Sifei Liu, Pavlo Molchanov, Ming-Hsuan Yang, and Jan Kautz. Scops: Self-supervised co-part segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 869–878, 2019. 2, 3, 6

[22] Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10124–10134, 2023. 2

[23] Hyung-Kwon Ko, Gwanmo Park, Hyeon Jeon, Jaemin Jo, Juho Kim, and Jinwook Seo. Large-scale text-to-image generation models for visual artists' creative works. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*, pages 919–933, 2023. 2

[24] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Semi-local affine parts for object recognition. In *British Machine Vision Conference, BMVC 2004, Kingston, UK, September 7-9, 2004. Proceedings*, pages 1–10. BMVA Press, 2004. 3

[25] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 6

[26] Shanglin Li, Bohan Zeng, Yutang Feng, Sicheng Gao, Xiuhui Liu, Jiaming Liu, Lin Li, Xu Tang, Yao Hu, Jianzhuang Liu, et al. Zone: Zero-shot instruction-guided local editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6254–6263, 2024. 7, 1

[27] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22511–22521, 2023. 1, 2, 3

[28] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. In

International Conference on Learning Representations, 2022. 7

[29] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 6

[30] Wan-Duo Kurt Ma, JP Lewis, W Bastiaan Kleijn, and Thomas Leung. Directed diffusion: Direct control of object placement through attention guidance. *arXiv preprint arXiv:2302.13153*, 2023. 2, 4

[31] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 2, 3

[32] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6038–6047, 2023. 6, 5

[33] Kam Woh Ng, Xiatian Zhu, Yi-Zhe Song, and Tao Xiang. Partcraft: Crafting creative objects by parts. *arXiv preprint arXiv:2407.04604*, 2024. 3, 7, 6

[34] Rishubh Parihar, Abhijnya Bhat, Abhipsa Basu, Saswat Mallick, Jogendra Nath Kundu, and R Venkatesh Babu. Balancing act: distribution-guided debiasing in diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6668–6678, 2024. 3

[35] Rishubh Parihar, Harsh Gupta, Sachidanand VS, and R Venkatesh Babu. Text2place: Affordance-aware text guided human placement. In *European Conference on Computer Vision*, pages 57–77. Springer, 2024. 3

[36] Rishubh Parihar, VS Sachidanand, Sabariswaran Mani, Tejan Karmali, and R Venkatesh Babu. Precisecontrol: Enhancing text-to-image diffusion models with fine-grained attribute control. In *European Conference on Computer Vision*, pages 469–487. Springer, 2024. 3

[37] Or Patashnik, Daniel Garibi, Idan Azuri, Hadar Averbuch-Elor, and Daniel Cohen-Or. Localizing object-level shape variations with text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 4, 5

[38] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 1, 2

[39] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 1

[40] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 2

[41] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 2

[42] 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. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 2, 8

[43] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2105.05233*, 2021. 6, 5

[44] Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Pontus Stenetorp, Jimmy Lin, and Ferhan Ture. What the daam: Interpreting stable diffusion using cross attention. *arXiv preprint arXiv:2210.04885*, 2022. 4

[45] Yoad Tewel, Omri Kaduri, Rinon Gal, Yoni Kasten, Lior Wolf, Gal Chechik, and Yuval Atzmon. Training-free consistent text-to-image generation. *arXiv preprint arXiv:2402.03286*, 2024. 3

[46] Junjiao Tian, Lavisha Aggarwal, Andrea Colaco, Zsolt Kira, and Mar Gonzalez-Franco. Diffuse, attend, and segment: Unsupervised zero-shot segmentation using stable diffusion. *arXiv preprint arXiv:2308.12469*, 2023. 2

[47] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1921–1930, 2023. 5, 7, 1

[48] Andrea Vedaldi, Siddharth Mahendran, Stavros Tsogkas, Subhransu Maji, Ross Girshick, Juho Kannala, Esa Rahtu, Iasonas Kokkinos, Matthew B. Blaschko, David Weiss, Ben Taskar, Karen Simonyan, Naomi Saphra, and Sammy Mohamed. Understanding objects in detail with fine-grained attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 3

[49] Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. $p+$: Extended textual conditioning in text-to-image generation. *arXiv preprint arXiv:2303.09522*, 2023. 3

[50] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. Caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 2, 6

[51] Zirui Wang, Zhizhou Sha, Zheng Ding, Yilin Wang, and Zhuowen Tu. Tokencompose: Grounding diffusion with token-level supervision. *arXiv preprint arXiv:2312.03626*, 2023. 3

[52] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-Vocabulary Panoptic Segmentation with Text-to-Image Diffusion Models. *arXiv preprint arXiv:2303.04803*, 2023. 2

[53] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei

Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2(3):5, 2022. 2

[54] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems*, 36:31428–31449, 2023. 7, 1

[55] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 1, 2, 3

[56] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 7