This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Zero-Painter: Training-Free Layout Control for Text-to-Image Synthesis

Marianna Ohanyan[1*] Hayk Manukyan[1*] Zhangyang Wang[1,2]
Shant Navasardyan[1] Humphrey Shi[1,3]
[1]Picsart AI Research (PAIR) [2]UT Austin [3]Georgia Tech
https://github.com/Picsart-AI-Research/Zero-Painter

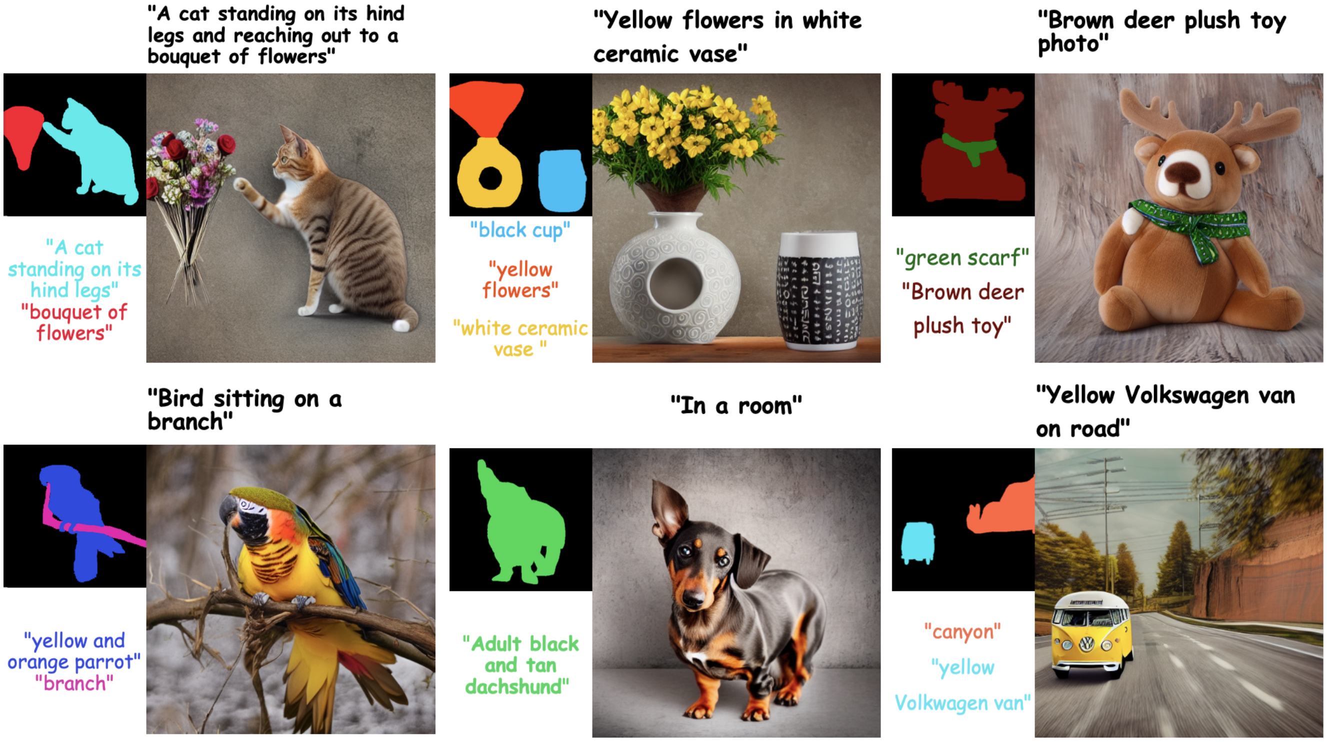

"A cat standing on its hind legs and reaching out to a bouquet of flowers"
"A cat standing on its hind legs"
"bouquet of flowers"
"Yellow flowers in white ceramic vase"
"black cup"
"yellow flowers"
"white ceramic vase "
"Brown deer plush toy photo"
"green scarf"
"Brown deer plush toy"
"Bird sitting on a branch"
"yellow and orange parrot"
"branch"
"In a room"
"Adult black and tan dachshund"
"Yellow Volkswagen van on road"
"canyon"
"yellow Volkwagen van"

Figure 1. Embark on a visual journey with Zero-Painter: a novel training-free framework for layout-conditional text-to-image generation. This new pipeline brings images to life using object masks and individual descriptions, seamlessly fused with a powerful global text prompt.

Abstract

We present Zero-Painter, a novel training-free framework for layout-conditional text-to-image synthesis that facilitates the creation of detailed and controlled imagery from textual prompts. Our method utilizes object masks and individual descriptions, coupled with a global text prompt, to generate images with high fidelity. Zero-Painter employs a two-stage process involving our novel Prompt-Adjusted Cross-Attention (PACA) and Region-Grouped Cross-Attention (ReGCA) blocks, ensuring precise alignment of generated objects with textual prompts and mask shapes. Our extensive experiments demonstrate that Zero-Painter surpasses current state-of-the-art methods in preserving textual details and adhering to mask shapes.

1. Introduction

Recent innovations in generative AI have revolutionized the creative landscape, allowing the generation of strikingly realistic images [35, 38, 42] or videos [4, 5, 14, 18] from text.

[*]Equal contribution.

However, crafting detailed prompts to guide every aspect of an image can be cumbersome and time-intensive. Furthermore, traditional text-to-image models often falter when faced with intricate prompts that describe multiple objects and their respective attributes. To address these challenges, layout-conditional text-to-image models have been developed that leverage additional inputs such as segmentation masks [1–3, 22] or bounding boxes [22, 28, 57] together with text. This approach facilitates the creation of images with precise attributes, granting artists and designers granular control over the visual components.

Early iterations of layout-conditional text-to-image methods [9, 50], employing GANs [11] and diffusion models [15, 44], achieved remarkable results using a closed vocabulary. However, their reliance on fixed class labels restricted their ability to prompt free-form attributes for the object in the layout. The introduction of GLIGEN [22] marked a significant advancement with its open-vocabulary and a multitude of new control mechanisms, including bounding box and text pairs, keypoints, edges, depth, and class-based segmentation maps. Later eDiff-I [2] introduced the Paint-With-Words approach, allowing open-vocabulary prompting with free-form mask control. Subsequently, MultiDiffusion [3] was introduced, capable of processing local prompts independently from the global prompt, adding flexibility. While these methods can generate visually convincing and prompt-aligned results, they are not always capable of keeping the shapes of the objects inside the mask. Despite this innovation, the absence of explicit mask conditioning often resulted in discrepancies between the shapes of generated objects and the provided masks.

To overcome these challenges, we present Zero-Painter, an innovative training-free method for layout-conditional text-to-image synthesis. It generates images from object masks and individual descriptions, alongside a global text prompt, as showcased in Fig. 1. Our process is bifurcated into two stages: initially, we generate individual objects, each endowed with unique attributes, using our Prompt-Adjusted Cross-Attention (PACA) module. These objects are then seamlessly integrated into a single scene through our Region-Grouped Cross-Attention (ReGCA) block. This ensures the generated objects not only align with the prompts but also conform to the shapes of the provided masks. Through rigorous testing, we have found that Zero-Painter surpasses state-of-the-art methods, particularly in maintaining the textual integrity of individual objects and adhering to the shapes of the given masks.

In summary, our contributions are as follows:

- We introduce Zero-Painter, a novel training-free framework for layout-conditional text-to-image synthesis, enabling the generation of objects with specified shapes and distinct attributes.
- We unveil the Prompt-Adjusted Cross-Attention (PACA) and Region-Grouped Cross-Attention (ReGCA) blocks, which significantly improve the shape fidelity and characteristic preservation of the generated objects.
- Comprehensive experiments validate Zero-Painter's superiority over existing state-of-the-art methods, as evidenced through both quantitative and qualitative comparisons.

2. Related Work

2.1. Text-to-Image Generation

Recently, significant advancements have occurred in the field of text-to-image generation. Approaches based on Generative Adversarial Networks (GANs) [12, 30, 53, 61] have yielded promising results in constrained domains. With the rise of transformer [49] models, zero-shot open-domain models were introduced. Notably, both Dall-E [34] and VQ-GAN [8] propose a two-stage approach. Initially, they employ a discrete Variational Autoencoder (VAE) [19, 37] to discover a comprehensive semantic space.

Later, Parti [60] illustrates the practicality of expanding autoregressive models in terms of scalability.

With the introduction of Diffusion-based models [38], the quality of text-to-image generation has significantly improved. DALL-E 2 [36] utilizes CLIP [32] for the text-to-image mapping process through diffusion mechanisms and trains a CLIP decoder. Furthermore, Imagen [42] leverages large pre-trained language models like T5 on textual data [33], achieving superior alignment between images and text, as well as enhanced sample fidelity. Lastly, eDiff-I [2] employs an expert-based approach, with different expert models handling generation at various timestep ranges.

2.2. Layout-to-Image Generation

Past research focused on image generation from structured layouts with fixed classes for content control [10, 13, 17, 25, 29, 46, 54]. CLIP [31] marked a paradigm shift by introducing zero-shot learning and enabling a transition from fixed to free-form text control. Recent advancements, including No-token-left-behind [28] and Gligen [22], proposed methods incorporating free-form text and bounding boxes, while Make-A-Scene [9] presented an innovative approach with a fixed set of labels but free-form mask-based control. Later approaches like Spa-text [1], eDiff-I [2], and MultiDiffusion [3] combine elements of free-form text and masks to broaden the scope of image generation.

2.3. Text-Guided Image Inpainting

The problem of image inpainting is known in the community and has been tackled in numerous works: [12, 21, 24, 26, 27, 27, 45, 55, 58, 59, 59, 62, 63]. With the recent rise of text-guided generative models, the text-guided version of image-inpainting has become relevant. Research in this field is progressing rapidly, and works like SmartBrush [52], Imagen Editor [51], and Uni-paint [56] present significant advancements. Most notably, Stable Inpainting [38] is a modification of the Stable Diffusion model, that is fine-tuned for text-guided inpainting. The base model is enhanced by concatenating the input image and inpainting mask as additional conditioning to the UNet model's latent input. The weights of the additional channels have been initialized with zeros, and fine tuned on the LAION [43] dataset using randomly generated inpainting masks.

3. Method

In this section we introduce Zero-Painter, a training-free layout-conditional text-to-image generation framework. First, we provide a brief background on diffusion models, focusing on Stable Diffusion (SD) [38]. Then, we present an overview of Zero-Painter, our two-stage pipeline that encompasses Single Object Generation (SOG) and Comprehensive Composition (CC). We delve into each stage, highlighting the design of Prompt-Adjusted Cross-Attention (PACA) and Region-Grouped Cross-Attention (ReGCA) layers for improved shape alignment and characteristic preservation.

3.1. Stable Diffusion

Stable Diffusion [38] is an LDM that works in the latent space of VQ-VAE [48] (or VQ-GAN [8] for the original LDM). During the diffusion process, Gaussian noise is iteratively added to the input latent tensor $x_0 \in \mathbb{R}^{h\times w\times c}$, such that the conditional distribution $q(x_t|x_{t-1})$ is:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I),\ t = 1,..,T \quad (1)$$

where $\{\beta_t\}_{t=1}^{T}$ are hyperparameters, and T is large enough that x_T becomes very close to $\mathcal{N}(0, I)$. By unraveling the process and denoting $\alpha_t = \prod_{i=1}^{t}(1-\beta_i)$ one can get a direct formula for x_t:

$$x_t = \sqrt{\alpha_t}x_0 + \sqrt{1-\alpha_t}\epsilon,\ \epsilon \sim N(0, 1). \quad (2)$$

The objective of SD is to learn a backward process

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)), \quad (3)$$

where $\mu_\theta(x_t, t), \Sigma_\theta(x_t, t)$ are parametric learnable functions (for simplicity usually $\Sigma_\theta(x_t, t) = \mathrm{diag}(\sigma^2)$ for a hyperparameter σ). Later, sampling $x_T \sim \mathcal{N}(0, I)$ and performing the backwards process for $t = T, \ldots, 1$ allows the generation of valid images, where final latent x_0 has to be decoded using the decoder $\mathcal{D}(x_0)$ of the VAE.

An alternative deterministic sampling approach was proposed in [44] called DDIM:

$$x_{t-1} = \sqrt{\alpha_{t-1}}\left(\frac{x_t - \sqrt{1-\alpha_t}\epsilon_\theta^t(x_t)}{\sqrt{\alpha_t}}\right) + \sqrt{1-\alpha_{t-1}}\epsilon_\theta^t(x_t), \quad t = T, \ldots, 1, \quad (4)$$

where

$$\epsilon_\theta^t(x_t) = \frac{\sqrt{1-\alpha_t}}{\beta_t}x_t + \frac{(1-\beta_t)(1-\alpha_t)}{\beta_t}\mu_\theta(x_t, t). \quad (5)$$

For Stable Diffusion the function $\epsilon_\theta^t(x_t, \tau)$ is directly predicted using a UNet. For text-to-image generation the UNet is altered by adding Cross-Attention layers, and an additional input textual prompt τ is provided:

$$x_{t-1} = \sqrt{\alpha_{t-1}}\left(\frac{x_t - \sqrt{1-\alpha_t}\epsilon_\theta^t(x_t, \tau)}{\sqrt{\alpha_t}}\right) + \sqrt{1-\alpha_{t-1}}\epsilon_\theta^t(x_t, \tau), \quad t = T, \ldots, 1. \quad (6)$$

3.2. Zero-Painter

The problem of layout-conditional text-to-image generation can be formulated as follows: Given a layout as a set of binary masks $M_i \in \{0, 1\}^{H\times W}$, $i = 1, \ldots, n$, indicating the shapes and positions of individual objects in a desired image, with corresponding textual prompts τ_i describing each object separately, as well as a global prompt τ_{global} describing the image in its entirety, the goal is to generate an output image $I \in \mathbb{R}^{H\times W}$ matching τ_{global} while containing the objects following the shapes and positions of the layout $\{M_i\}_{i=1}^{n}$, and the prompts $\{\tau_i\}_{i=1}^{n}$.

To this end, Zero-Painter introduces an optimization-free two-stage pipeline, enabling the independent generation of objects followed by their seamless composition into a single image. This two-stage approach gives an advantage of utilizing the whole capacity of a diffusion model on single object generation resulting in better shape and characteristics alignment than current one-stage methods (see Fig. 9).

During the first stage, we leverage PACA layer to individually generate images I_i, $i = 1, \ldots, n$, each containing a single object on a flat background that follows the shape/position of the binary mask M_i and matches the description τ_i (see Fig. 2). The use of flat backgrounds aids in the easy identification and segmentation of the generated objects, especially if they differ slightly from the original mask.

The second stage takes the generated images $I_1, \ldots, I_n$, and combines them according to the global prompt τ_{global} and the individual mask-prompt pairs (M_i, τ_i). To coherently combine the generated objects, we first separate them from their backgrounds by utilizing the Segment Anything Model (SAM) [20] and obtaining new object masks $M_i' \in$

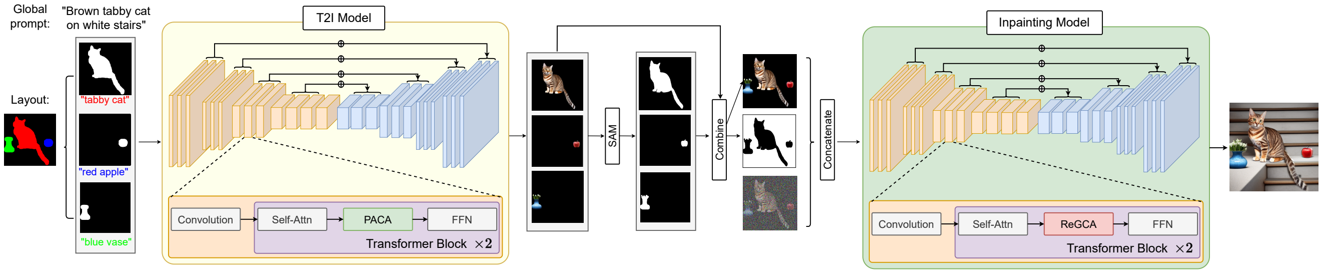


Figure 2. Optimization-Free Two-Stage Pipeline for Zero-Shot Image Composition: (a) In the first stage, we focus on single object generation, leveraging the innovative Prompt-Adjusted Cross-Attention (PACA) layer. (b) Moving to the comprehensive composition stage, we introduce the Region-Grouped Cross-Attention (ReGCA) block, facilitating seamless and dynamic composition of generated objects.

$\{0,1\}^{H\times W}$. Note that using the initial mask M_i for separating the foreground object in I_i may cause undesirable object cuts if the generated object shape has a slight mismatch with the given layout mask M_i (see Fig. 7). Then we leverage Stable Inpainting [38] modified with our Region-Grouped Cross-Attention (ReGCA) module to seamlessly combine the generated objects and fill the background region indicated by the mask $M'_{bg} = (1 - M'_1) \odot \ldots \odot (1 - M'_n)$. The inpainting is done with the textual guidance of τ_{global}.

3.3. Single Object Generation (SOG)

The goal of the SOG stage is to produce an image from a binary mask M_i and textual description τ_i, ensuring the generated object's shape matches M_i and its description matches τ_i. The rest of the image is filled with a flat background for easier separation of the generated objects later.

We use a pre-trained Stable Diffusion (SD) [38] model to generate an image with a specified pair (M_i,τ_i). [2] shows that the high timesteps in the diffusion process are mostly responsible for creating the object silhouette, while later steps create details and refine the object. Since in our case the target shape is known and described by M_i, we start the diffusion backward process from an intermediate timestep $T' < T$, and provide the shape information through a starting latent $x_{T'}$ obtained using the mask M_i. To construct $x_{T'}$ we consider two factors: (i) the background of the final image should be of a flat color; (ii) the foreground object should be constrained to M_i. We first obtain the latent code of the flat background corresponding to timestep T' by applying noise on a latent encoding of a constant black image:

$$x^{\text{flat}}_{T'} = \sqrt{\alpha_{T'}}\mathcal{E}(I^{\text{flat}}) + \sqrt{1-\alpha_{T'}}\epsilon, \tag{7}$$

where I^{flat} is the constant black image, $\mathcal{E}()$ is the VAE encoder and $\epsilon \sim \mathcal{N}(\mathbf{0},\mathbf{1})$ is sampled from the standard gaussian distribution.

We initialize the foreground region M_i of $x_{T'}$ from a sampled Gaussian noise $\epsilon \sim N(\mathbf{0},\mathbf{1})$. While this deviates from the value expected according to Eq. (2), we find that providing any specific value for x_0 may encourage the model to generate an image similar to x_0, introducing unnecessary characteristics. Furthermore, for sufficiently high values of T' the noise to signal ratio is so high, that ϵ becomes a good enough approximation, and the deviation remains within the tolerance of the model. To further mitigate potential quality loss, we add a refinement sub-stage to object composition stage of Zero-Painter, aiming to correct any remaining errors in the generated image.

In summary, the starting latent $x_{T'}$ for the single object generation stage is defined as

$$x_{T'} = (1 - M_i) \odot x^{\text{flat}}_{T'} + M_i \odot \epsilon, \tag{8}$$

where M_i is the mask, and $\epsilon \sim \mathcal{N}(\mathbf{0},\mathbf{1})$ is randomly sampled from the standard gaussian distribution.

After we get the initial latent $x_{T'}$ we apply the DDIM backward process for $t = T',\ldots,1$ by leveraging SD, enhanced with our Prompt-Aware Cross-Attention (PACA) layers designed for the individual object shape alignment with M_i and coherence with the prompt τ_i:

$$\begin{aligned} x^{\text{pred}}_0(t) &= \frac{x_t - \sqrt{1-\alpha_t}\epsilon^t_\theta(x_t)}{\sqrt{\alpha_t}}, \\ x_{t-1} &= \sqrt{\alpha_{t-1}}x^{\text{pred}}_0(t) + \sqrt{1-\alpha_{t-1}}\epsilon^t_\theta(x_t), \end{aligned} \tag{9}$$

where $\epsilon^t_\theta()$ is SD augmented with PACA layers.

To further ensure that the object does not extend outside of the masked area, we blend the noised latent of the flat background x^{flat}_t and the predicted x_t at every timestep $t = T',\ldots,0$:

$$x_t = M_i \odot x_t + (1 - M_i) \odot x^{\text{flat}}_t, \tag{10}$$

3.3.1 Prompt-Aware Cross-Attention (PACA)

The PACA layer (Fig. 5) plays a crucial role in the SOG process. It ensures accurate object generation based on textual descriptions and masks, while preventing generation

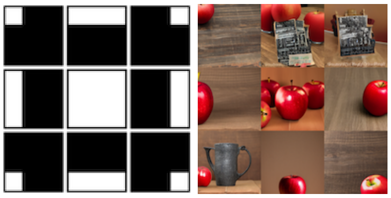
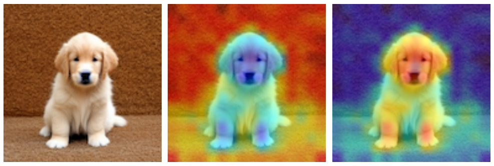


Figure 3. Effect of the SOT token. The similarity with the SOT token has been increased during text-to-image generation (at every step) in the non-white areas of the masks (left side). Prompt: ”photo of a red apple, centered”.



(a) Output (b) SOT Token (c) Other Tokens

Figure 4. Similarity of the SOT token vs all other tokens combined.

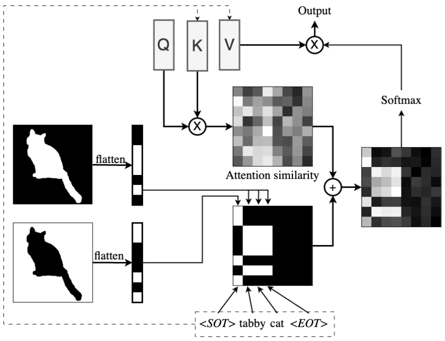


Figure 5. Overview of Prompt-Aware Cross-Attention(PACA) during the Invdvidual Object Generation stage.

outside the masked area. For our PACA layer we employ a mechanism similar to eDiff-I [2], i.e. by modifying the cross-attention similarity matrix S. To encourage the generation of the object inside the masked area, we increase similarity values of queries $q_j, j \in M_i$ corresponding to the masked area M_i, with the keys of all prompt tokens excluding the SOT. We exclude the SOT since we noticed that during vanilla text-to-image generation increasing the similarity of a pixel with the SOT token results in the output pixel becoming a generic background pixel (see Figs. 3 and 4). For the same reason, we increase the similarity values of non-masked pixel $q_j, j \notin M_i$ with the SOT token. Therefore, the similarity matrix S of selected cross attention layers is modified as follows:

$$S'_j = \begin{cases} S_j + w_t \sum_{k=1}^{N} \mathbb{1}_k & \text{if } j \in M_i \\ S_j + w_t \mathbb{1}_0 & \text{otherwise} \end{cases} \tag{11}$$

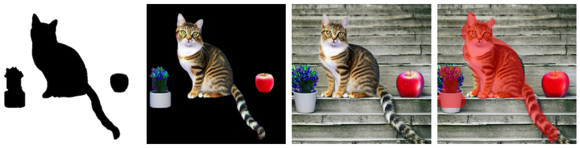


Figure 6. Left to right: the cobined mask used for inpainting, the combined input image, the resulting comprehensive composition and an overlay with original input layout.

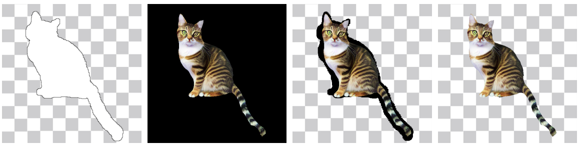


Figure 7. Example to illustrate importance of SAM. Left to right: original mask, output of Single Object Generation, output cropped using the original mask, output cropped using the mask adjusted by SAM.

where S_j is the column of the similarity matrix corresponding to pixel with index j, $\mathbb{1}_k = [0...1...0]$ is an indicator vector and N is the index of the EOT token, accordingly, index 0 is the SOT token. Inspired by [2] we choose $w_t = w' \log(1+\sigma_t) \max{(QK^T)}$, where σ_t is the noise-to-signal ratio and w' is a hyperparameter.

3.4. Comprehensive Composition (CC)

The CC phase aims to combine all previously generated objects into a single image that fits the description of the global prompt τ_{global}. In 3.4.1 we describe how to perform object segmentation for each individual object. In 3.4.2 - the process of inpainting the background region and in 3.4.3 - details concerning Region-Grouped Cross-Attention (ReGCA).

3.4.1 Object Segmentation

Objects from the SOG process may not precisely follow the input mask M_i, especially with hand-drawn masks. Extracting objects using the original mask may introduce background segments in the CC stage (see Fig. 7). To address this, we perform object segmentation on the output images, completely separating the generated object from the image. Using a pre-trained Segment-Anything-Model [20] with the bounding box of the original mask as input, we find the intersection of the output and original masks.

$$\hat{M}_i = \text{SAM}(x_0 | bbox(M_i)) * M_i \tag{12}$$

where $bbox(M_i)$, is the minimal bounding box spanning the mask M_i (in $[x, y, w, h]$ format).

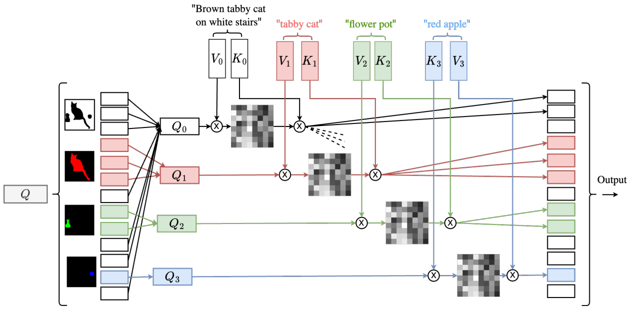


Figure 8. Region-Grouped Cross-Attention (ReGCA) architecture.

3.4.2 Inpainting

We combine all the components using a pre-trained Stable Inpainting model [38]. Rather than creating a new object in the masked region, we utilize the model to synthesize a background around the existing ones. We construct the input image for the inpainting model by combining x_0^i generated during SOG using predicted masks $\hat{M}_i$: $x_{\text{known}} = \sum_i x_0^i \hat{M}_i$. Similarly we define the combined mask: $M = 1 - \sum_i \hat{M}_i$.

Similar to Sec. 3.3, we choose a starting step $T'' < T$, discouraging the model from generating new objects, and prompting it to focus on generating a background. Additionally, to better maintain the structural coherence of pre-existing objects, we initialize the initial latent noise $x_{T''}$ within the known region as the noised latent $x_{T''}^{\text{known}}$ of the input image x^{known}.

$$x_{T''}^{\text{known}} = \sqrt{\alpha_{t-1}} x^{\text{known}} + \sqrt{1 - \alpha_{t-1}} \epsilon_1 \tag{13}$$

$$x_{T''} = M x_{T''}^{\text{known}} + (1 - M)\epsilon_2 \tag{14}$$

where ϵ_1 and ϵ_2 are random noise vectors sampled from $N(0, 1)$

We perform DDIM iterations from timestep T'' down to a minimum timestep t_{min} using the mask M. For the last t_{min} timesteps, we increase the mask to cover the entire image, allowing the model to fine-tune the known region together with the newly generated regions, to obtain a more homogeneous final image.

3.4.3 Region-Grouped Cross-Attention (ReGCA)

Similar to PACA, the ReGCA (Fig. 8) layer is crucial for obtaining a coherent image after inpainting. We modify cross-attention layers for two purposes: first, we use negative prompts to prohibit the model from generating existing objects outside the known region. Second, we ensure the model receives sufficient information about existing objects through cross-attention values, even when the corresponding object prompts are missing from the global caption. To achieve this, we divide the pixels of the latent vector into groups, based on which object, or background they belong to (see Fig. 8). For the object with index o, mask $\hat{M}_o$ and prompt T_o we select the subset of queries $q_i^o = \{q_i | i \in \hat{M}_o\}$. For each group we compute its own set of key-value pairs k_j^o, v_j^o, as well as their unconditional counterparts $\tilde{k}_j^o$, $\tilde{v}_j^o$. k_j^o, v_j^o are computed from the tokens of the object prompt T_o, while $\hat{K}^k$ and $\hat{V}^k$ - using an empty string.

We add an additional group for the pixels belonging to the background $q_i^{\text{bg}} = \{q_i | i \notin \hat{M}_o \forall o\}$. For this group, we use the global prompt τ_{global} for computing k_j^{bg} and v_j^{bg}, while for $\tilde{k}_j^{\text{bg}}$, $\tilde{v}_j^{\text{bg}}$ we construct a new prompt from the comma separated concatenation of all object prompts $T_{\text{bg}}^{\text{uc}} = \cup T_o$. Using a non-empty prompt with the unconditional model serves as a negative prompt, preventing the generation of existing objects in the background area.

After individually computing the cross attention outputs for each group, they are combined into a single output by putting pixels from each output into their corresponding positions in the original input.

4. Experiments

4.1. Implementation details

Our implementation is based on the ”Stability-AI” GitHub repository [41]. We use pre-trained weights of Stable Diffusion 1.4 [39] and Stable-Inpainting 1.5 [40] from the huggingface repository. For starting timesteps we use $T', T'' = 800$ and $t_{min} = 100$. We use 40 DDIM steps (we skip 10 steps due to $T' = 800$).

4.2. Quantitative Results

To assess our model and compare it with other state-of-the-art models, we created a validation set consisting of 3000 MSCOCO [23] segmentation layouts. We filtered out masks that have an area $< 5\%$ of the image, and resize all layouts to 512x512.

We compared our model with existing layout2image approaches, that have publicly available repositories: eDiff-I using Stable Diffusion 1.4 [2, 6], Multidiffusion [3, 47], as well as bounding-box based approaches: Gligen [7, 22] and NTLB [16, 28]. For the latter, we used the bounding boxes of the corresponding masks as input. We use Local CLIP-Score to compute text-alignment: we crop the image using bounding boxes of each mask and compute the CLIP Score with the corresponding object prompt.

$$\mathcal{S}_{\text{CLIP}}(I, C) = \frac{1}{n} \sum_{i=1}^{n} \max\left(100 \cdot \cos\left(E_{\text{img}}\left(I_i^{\text{crop}}\right), E_{\text{txt}}(\tau_i)\right), 0\right) \tag{15}$$

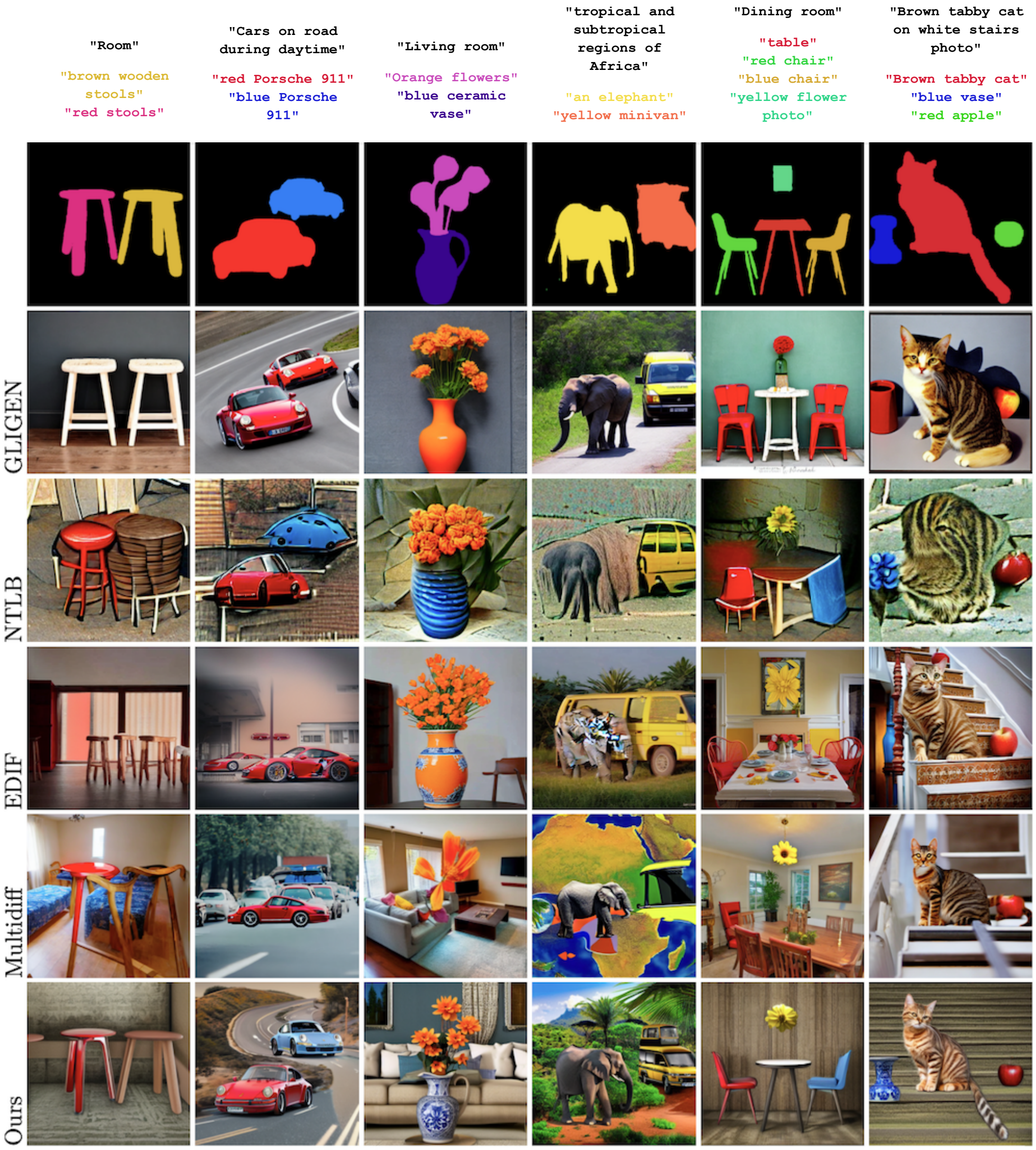


Figure 9. Qualitative comparison between the Zero-Painter pipeline and state-of-the-art models.

where τ_i is the object prompt, I_i^{crop} is the cropped region of image I using the bounding box of M_i and E_{img}, E_{txt} correspond to extracting CLIP embedding. We measure shape alignment using the average of Local Intersection over Union (IoU). Utilizing a pre-trained SAM model [20] on the cropped region, we determine the shape of the final generated object and compute its IoU with the original mask.

$$\mathcal{S}_{\text{IoU}} = \frac{1}{n}\sum_{i=1}^{n} \text{IoU}(\text{SAM}(I|\text{box}(M_i)), M_i) \tag{16}$$

where IoU = $\frac{\text{Area of Overlap}}{\text{Area of Union}}$, I is generated image.

As evident from Tab. 1 Zero-Painter outperforms other state-of-the-art approaches.

4.3. Qualitative Results

To present a more detailed and visual comparison of our model, we hand-crafted a smaller test-set using images from Unsplash as the base of our layouts. For a more extensive comparison with a larger image set, see the Appendix. Meanwhile, Fig. 9 displays a subset of examples.

As mentioned above, the most common issue for competitor models is the leakage of properties between different objects, e.g. the colors of the chair in column 1, colors

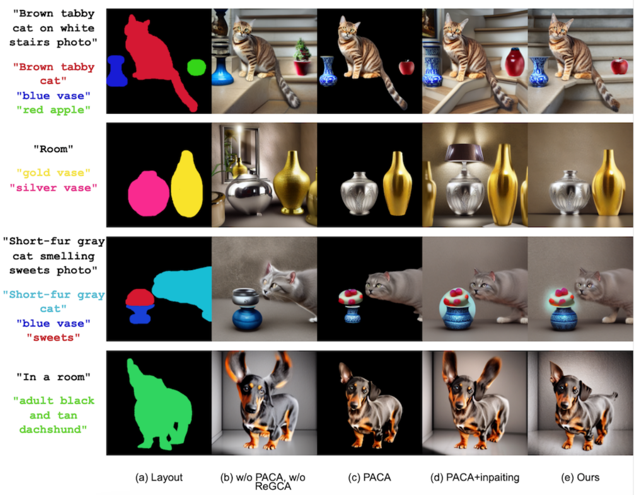


Figure 10. Ablation study.

of the cars in column 2. In addition, some objects can be completely neglected, like the vase in column 6 and 3 for Multidiff. We also note, that this particular implementation of EDIFF-I might generate visual artifacts like in column 2 and 4. Similarly, Multidiff's outputs can sometimes look ”mutated”, like the chairs in column 1.

In comparison, our model is less likely to struggle from all aforementioned issues. Since the objects are generated individually using PACA module and later refined with ReGCA, the properties of each object are much better preserved in the final image.

5. Ablation Study

In this section, we assess the significance of two presented modules PACA and ReGCA. Results without PACA and ReGCA modules are presented in Fig (column 2). PACA ensures that the generated object tries to fill as much of the mask as possible Fig (column 3). Moreover by increasing the similarity with all tokens in the object prompt, PACA ensures that the object properties are kept as closely as possible. For instance, consider the red apple in the 1st row, 3rd column, which is generated using PACA. Notably, in the 1st row, 2nd column, where ”red apple” is intended, the model generated something similar to a ”red vase”. The impact of PACA becomes more evident in the 3rd row, where, in its absence, the model generates a gray blob instead of the intended ”candies.” This deviation highlights PACA's crucial role in maintaining coherence between generated content and textual description.

The CC stage faces challenges even when objects are accurately generated in the PACA stage. When using a basic inpainting pipeline, two issues arise. First, some objects extend beyond their intended boundaries, disrupting visual coherence (column 4, see the dog's ears in row 4 and the vase in the 1st row). Second, inpainting with τ_{global} fails when it lacks information about all objects, as seen in the first row where the absence of the ”apple” prompt results in a failed inpainting. ReGCA in column 5 prevents shape continuation, addressing limitations due to missing object prompts in τ_{global}.

Model	EDIFF-I	GLIGEN	NTLB	MDF	Ours
CLIP (local)	25.3	25.52	25.71	<u>26.10</u>	**26.68**
IoU (local)	<u>0.62</u>	N/A	0.50	0.58	**0.75**

Table 1. Quantitative comparison.

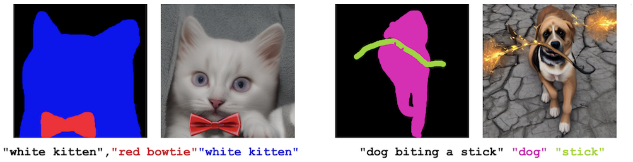

"white kitten","red bowtie""white kitten"
"dog biting a stick" "dog" "stick"

Figure 11. Zero-Painter's limitation in handling overlapping masks.

6. Limitations

While Zero-Painter excels in generating detailed and controlled images, there are still limitations. One such limitation arises when dealing with overlapping masks. In instances where masks intersect or overlap, the resulting images may exhibit unnatural or less visually coherent outcomes (see Fig. 11). This challenge occurs during the CC stage: since we are using ReGCA for inpainting only the background region, objects inside masks remain unchanged. Although Zero-Painter excels in many scenarios, improving this limitation is an area for future enhancement.

7. Conclusion

To this end, we propose Zero-Painter that is a training-free framework for layout-conditional text-to-image synthesis. The method utilizes object masks, individual descriptions, and a global text prompt, employing a two-stage process with novel Prompt-Adjusted Cross-Attention (PACA) and Region-Grouped Cross-Attention (ReGCA) blocks. These advancements ensure precise alignment of generated objects with textual prompts and mask shapes. Extensive experiments demonstrate that Zero-Painter's ability to preserve textual details and keeping shapes are superior to all existing methods. The paper introduces innovative contributions, including the novel framework, PACA and ReGCA blocks, and comprehensive experimental validation.

References

[1] Omri Avrahami, Thomas Hayes, Oran Gafni, Sonal Gupta, Yaniv Taigman, Devi Parikh, Dani Lischinski, Ohad Fried, and Xi Yin. Spatext: Spatio-textual representation for controllable image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18370–18380, 2023. 2

[2] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, Tero Karras, and Ming-Yu Liu. ediff-i: Text-to-image diffusion models with ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 2, 4, 5, 6

[3] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. *arXiv preprint arXiv:2302.08113*, 2023. 2, 6

[4] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, and et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv.org*, 2023. 1

[5] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 1

[6] cloneofsimo. Paint with words sd, 2023. 6

[7] GLIGEN Contributors. GLIGEN: Open-Set Grounded Text-to-Image Generation. https://github.com/gligen/GLIGEN, 2023. 6

[8] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 2, 3

[9] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors, 2022. 2

[10] Vidit Goel, Elia Peruzzo, Yifan Jiang, Dejia Xu, Nicu Sebe, Trevor Darrell, Zhangyang Wang, and Humphrey Shi. Pair-diffusion: Object-level image editing with structure-and-appearance paired diffusion models. *arXiv preprint arXiv:2303.17546*, 2023. 2

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 2

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 2

[13] S. He, W. Liao, M. Yang, Y. Yang, Y. Song, B. Rosenhahn, and T. Xiang. Context-aware layout to image generation with enhanced object appearance. *arXiv preprint arXiv:2103.11897*, 2021. 2

[14] Roberto Henschel, Levon Khachatryan, Daniil Hayrapetyan, Hayk Poghosyan, Vahram Tadevosyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Streamingt2v: Consistent, dynamic, and extendable long video generation from text. *arXiv preprint arXiv:2403.14773*, 2024. 1

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2

[16] Apple Inc. ml-no-token-left-behind. https://github.com/apple/ml-no-token-left-behind, 2023. 6

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

[18] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. *arXiv preprint arXiv:2303.13439*, 2023. 1

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

[20] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 3, 5, 7

[21] Jingyuan Li, Ning Wang, Lefei Zhang, Bo Du, and Dacheng Tao. Recurrent feature reasoning for image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7760–7768, 2020. 2

[22] 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. 2023. 2, 6

[23] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Doll'a r, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014. 6

[24] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European conference on computer vision (ECCV)*, pages 85–100, 2018. 2

[25] K. Ma and B. Zhao. Attribute-guided image generation from layout. *arXiv preprint arXiv:2008.11932*, 2020. 2

[26] Hayk Manukyan, Andranik Sargsyan, Barsegh Atanyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Hd-painter: High-resolution and prompt-faithful text-guided image inpainting with diffusion models. *arXiv preprint arXiv:2312.14091*, 2023. 2

[27] Shant Navasardyan and Marianna Ohanyan. The family of onion convolutions for image inpainting. *International Journal of Computer Vision*, pages 1–30, 2022. 2

[28] Roni Paiss, Hila Chefer, and Lior Wolf. No token left behind: Explainability-aided image classification and generation, 2022. 2, 6

[29] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2

[30] Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. Mirrorgan: Learning text-to-image generation by redescription. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1505–1514, 2019. 2

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 2

[32] 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 *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2

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

[34] 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. 2

[35] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022. *URL https://arxiv.org/abs/2204.06125*, 7, 2022. 1

[36] 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. 2

[37] Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with VQ-VAE-2. In *Advances in Neural Information Processing Systems*, pages 14825–14836, 2019. 2

[38] 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, 3, 4, 6

[39] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. Compvis stable diffusion v1.4. `https://huggingface.co/CompVis/stable-diffusion-v1-4`, 2023. November 2023. 6

[40] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. Runwayml stable diffusion v1.5. `https://huggingface.co/runwayml/stable-diffusion-v1-5`, 2023. November 2023. 6

[41] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. Stable diffusion version 2. `https://github.com/Stability-AI/stablediffusion`, Year Accessed. November 2023. 6

[42] 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. 1, 2

[43] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 3

[44] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2, 3

[45] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2149–2159, 2022. 2

[46] Tristan Sylvain, Pengchuan Zhang, Yoshua Bengio, R. Devon Hjelm, and Shikhar Sharma. Object-centric image generation from layouts. In *AAAI Conference on Artificial Intelligence*, 2020. 2

[47] Omer Bar Tal. MultiDiffusion: Fusing Diffusion Paths for Controlled Image Generation. `https://github.com/omerbt/MultiDiffusion`, 2023. 6

[48] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 3

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2

[50] Bo Wang, Tao Wu, Minfeng Zhu, and Peng Du. Interactive image synthesis with panoptic layout generation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7773–7782, 2022. 2

[51] S. Wang et al. Imagen editor and editbench: Advancing and evaluating text-guided image inpainting. *arXiv preprint arXiv:2212.06909*, 2022. 3

[52] S. Xie et al. Smartbrush: Text and shape guided object inpainting with diffusion model. *arXiv preprint arXiv:2212.05034*, 2022. 3

[53] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1316–1324, 2018. 2

[54] Xingqian Xu, Jiayi Guo, Zhangyang Wang, Gao Huang, Irfan Essa, and Humphrey Shi. Prompt-free diffusion: Taking”

text" out of text-to-image diffusion models. *arXiv preprint arXiv:2305.16223*, 2023. 2

[55] Xingqian Xu, Shant Navasardyan, Vahram Tadevosyan, Andranik Sargsyan, Yadong Mu, and Humphrey Shi. Image completion with heterogeneously filtered spectral hints. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023. 2

[56] Shiyuan Yang, Xiaodong Chen, and Jing Liao. Uni-paint: A unified framework for multimodal image inpainting with pretrained diffusion model. In *Proceedings of the 31st ACM International Conference on Multimedia*, page 3190–3199, New York, NY, USA, 2023. Association for Computing Machinery. 3

[57] Z. Yang, D. Liu, C. Wang, and J. Yang. Modeling image composition for complex scene generation. *arXiv:2206.00923*, 2022. 2

[58] Zili Yi, Qiang Tang, Shekoofeh Azizi, Daesik Jang, and Zhan Xu. Contextual residual aggregation for ultra high-resolution image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7508–7517, 2020. 2

[59] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4471–4480, 2019. 2

[60] 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*, 2022. 2

[61] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 5907–5915, 2017. 2

[62] Shengyu Zhao, Jonathan Cui, Yilun Sheng, Yue Dong, Xiao Liang, Eric I Chang, and Yan Xu. Large scale image completion via co-modulated generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2021. 2

[63] Haitian Zheng, Zhe Lin, Jingwan Lu, Scott Cohen, Eli Shechtman, Connelly Barnes, Jianming Zhang, Ning Xu, Sohrab Amirghodsi, and Jiebo Luo. Cm-gan: Image inpainting with cascaded modulation gan and object-aware training. *arXiv preprint arXiv:2203.11947*, 2022. 2