SpaText: Spatio-Textual Representation for Controllable Image Generation

Omri Avrahami[1,2] Thomas Hayes[1] Oran Gafni[1] Sonal Gupta[1]
Yaniv Taigman[1] Devi Parikh[1] Dani Lischinski[2] Ohad Fried[3] Xi Yin[1]

[1]Meta AI [2]The Hebrew University of Jerusalem [3]Reichman University

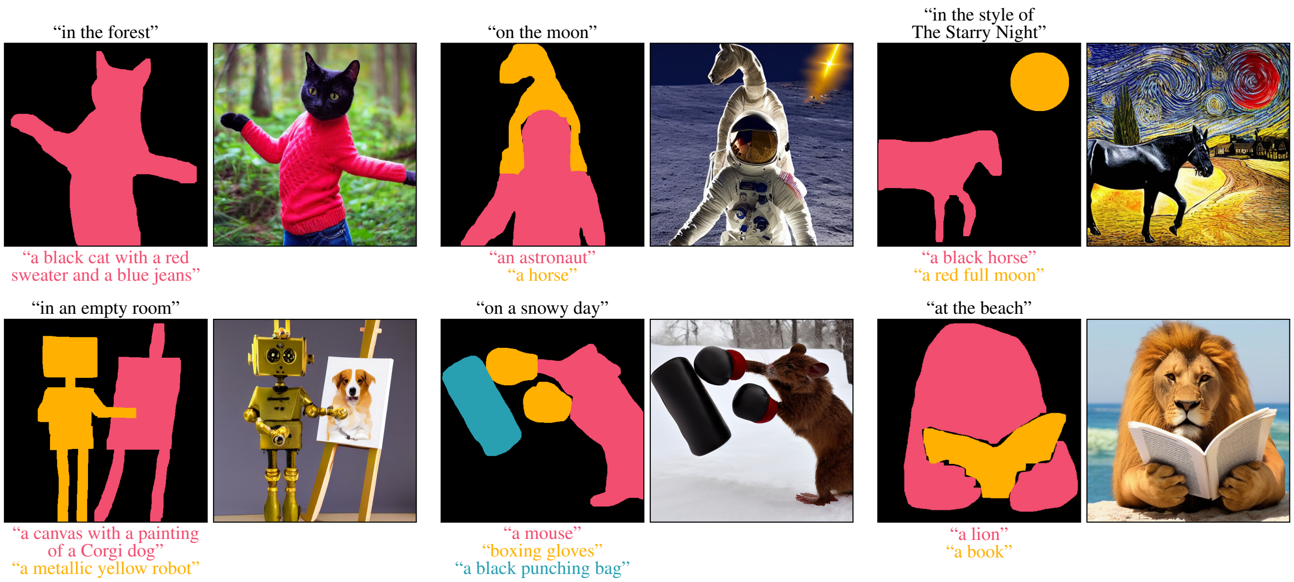

Figure 1. Samples of generated images from input text and our proposed spatio-textual representations. Each pair consists of an (i) input global text (top left, black), a spatio-textual representation describing each segment using free-form text prompts (left, colored text and sketches), and (ii) the corresponding generated image (right). As can be seen, SpaText is able to generate high-quality images that correspond to both the global text and spatio-textual representation content. (The colors are for illustration purposes only, and do not affect the actual inputs.)

Abstract

Recent text-to-image diffusion models are able to generate convincing results of unprecedented quality. However, it is nearly impossible to control the shapes of different regions/objects or their layout in a fine-grained fashion. Previous attempts to provide such controls were hindered by their reliance on a fixed set of labels. To this end, we present SpaText — a new method for text-to-image generation using open-vocabulary scene control. In addition to a global text prompt that describes the entire scene, the user provides a segmentation map where each region of interest is annotated by a free-form natural language description. Due to lack of large-scale datasets that have a detailed textual description for each region in the image, we choose to leverage the current large-scale text-to-image datasets and base our approach on a novel CLIP-based spatio-textual representation, and show its effectiveness on two state-of-the-art diffusion models: pixel-based and latent-based. In addition, we show how to extend the classifier-free guidance method in diffusion models to the multi-conditional case and present an alternative accelerated inference algorithm. Finally, we offer several automatic evaluation metrics and use them, in addition to FID scores and a user study, to evaluate our method and show that it achieves state-of-the-art results on image generation with free-form textual scene control.

1. Introduction

Imagine you could generate an image by dipping your digital paintbrush (so to speak) in a "black horse" paint, then sketching the specific position and posture of the horse, afterwards, dipping it again in a "red full moon" paint and sketching it the desired area. Finally, you want the entire image to be in the style of The Starry Night. Current state-of-the-art text-to-image models [51, 59, 72] leave much to

Project page is available at: https://omriavrahami.com/spatext be desired in achieving this vision.

The text-to-image interface is extremely powerful — a single prompt is able to represent an infinite number of possible images. However, it has its cost — on the one hand, it enables a novice user to explore an endless number of ideas, but, on the other hand, it limits controllability: if the user has a mental image that they wish to generate, with a specific layout of objects or regions in the image and their shapes, it is practically impossible to convey this information with text alone, as demonstrated in Figure 2. In addition, inferring spatial relations [72] from a single text prompt is one of the current limitations of SoTA models.

Make-A-Scene [22] proposed to tackle this problem by adding an additional (optional) input to text-to-image models, a *dense* segmentation map with *fixed* labels. The user can provide two inputs: a text prompt that describes the entire scene and an elaborate segmentation map that includes a label for each segment in the image. This way, the user can easily control the layout of the image. However, it suffers from the following drawbacks: (1) training the model with a fixed set of labels limits the quality for objects that are not in that set at inference time, (2) providing a dense segmentation can be cumbersome for users and undesirable in some cases, e.g., when the user prefers to provide a sketch for only a few main objects they care about, letting the model infer the rest of the layout; and (3) lack of fine-grained control over the specific characteristic of each instance. For example, even if the label set contains the label "dog", it is not clear how to generate several instances of dogs of different breeds in a single scene.

In order to tackle these drawbacks, we propose a different approach: (1) rather than using a fixed set of labels to represent each pixel in the segmentation map, we propose to represent it using *spatial free-form text*, and (2) rather than providing a dense segmentation map accounting for each pixel, we propose to use a *sparse* map, that describes only the objects that a user specifies (using spatial free-form text), while the rest of the scene remains unspecified. To summarize, we propose a new problem setting: given a *global text prompt* that describes the entire image, and a spatio-textual scene that specifies for segments of interest their *local text description* as well as their *position and shape*, a corresponding image is generated, as illustrated in Figure 1. These changes extend expressivity by providing the user with more control over the regions they care about, leaving the rest for the machine to figure out.

Acquiring a large-scale dataset that contains free-form textual descriptions for each segment in an image is prohibitively expensive, and such large-scale datasets do not exist to the best of our knowledge. Hence, we opt to extract the relevant information from existing image-text datasets. To this end, we propose a novel CLIP-based [49] spatio-textual representation that enables a user to specify for each segment its description using free-form text and its position and shape. During training, we extract *local regions* using a pre-trained panoptic segmentation model [69], and use them as input to a CLIP image encoder to create our representation. Then, at inference time, we use the *text descriptions* provided by the user, embed them using a CLIP text encoder, and translate them to the CLIP image embedding space using a prior model [51].

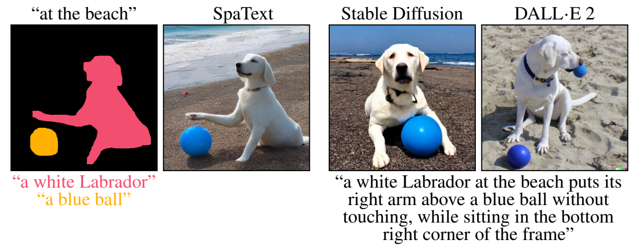

Figure 2. **Lack of fine-grained spatial control:** A user with a specific mental image of a Labrador dog holding its paw above a blue ball without touching, can easily generate it with a SpaText representation (left) but will struggle to do so with traditional text-to-image models (right) [52, 56].

In order to assess the effectiveness of our proposed representation SpaText, we implement it on two state-of-the-art types of text-to-image diffusion models: a pixel-based model (DALL·E 2 [51]) and a latent-based model (Stable Diffusion [56]). Both of these text-to-image models employ classifier-free guidance [33] at inference time, which supports a single conditioning input (text prompt). In order to adapt them to our multi-conditional input (global text as well as the spatio-textual representation), we demonstrate how classifier-free guidance can be extended to any multi-conditional case. To the best of our knowledge, we are the first to demonstrate this. Furthermore, we propose an additional, faster variant of this extension that trades-off controllability for inference time.

Finally, we propose several automatic evaluation metrics for our problem setting and use them along with the FID score to evaluate our method against its baselines. In addition, we conduct a user-study and show that our method is also preferred by human evaluators.

In summary, our contributions are: (1) we address a new scenario of image generation with free-form textual scene control, (2) we propose a novel spatio-textual representation that for each segment represents its semantic properties and structure, and demonstrate its effectiveness on two state-of-the-art diffusion models — pixel-based and latent-based, (3) we extend the classifier-free guidance in diffusion models to the multi-conditional case and present an alternative accelerated inference algorithm, and (4) we propose several automatic evaluation metrics and use them to compare against baselines we adapted from existing methods. We also evaluate via a user study. We find that our method achieves state-of-the-art results.

2. Related Work

Text-to-image generation. Recently, we have witnessed great advances in the field of text-to-image generation. The seminal works based on RNNs [13, 34] and GANs [25] produced promising low-resolution results [54, 71, 73, 74] in constrained domains (e.g., flowers [45] and birds [68]). Later, zero-shot open-domain models were achieved using transformer-based [67] approaches: DALL·E 1 [52] and VQ-GAN [20] propose a two-stage approach by first training a discrete VAE [37,53,66] to find a rich semantic space, then, at the second stage, they learn to model the joint distribution of text and image tokens autoregressively. CogView [18, 19] and Parti [72] also utilized a transformer model for this task. In parallel, diffusion based [15, 17, 32, 44, 60] text-to-image models were introduced: Latent Diffusion Models (LDMs) [56] performed the diffusion process on a lower-dimensional latent space instead on the pixel space. DALL·E 2 [51] proposed to perform the diffusion process on the CLIP_{img} space. Finally, Imagen [59] proposed to utilize a pre-trained T5 language model [50] for conditioning a pixel-based text-to-image diffusion model. Recently, retrieval-based models [3,8,12,57] proposed to augment the text-to-image models using an external database of images. All these methods do not tackle the problem of image generation with free-form textual scene control.

Scene-based text-to-image generation. Image generation with scene control has been studied in the past [21, 30, 31, 35, 40, 48, 55, 61–63, 75], but not with general masks and free-form text control. No Token Left Behind [46] proposed to leverage explainability-based method [10, 11] for image generation with spatial conditioning using VQGAN-CLIP [16] optimization. In addition, Make-A-Scene [22] proposed to add a *dense* segmentation map using a *fixed* set of labels to allow better controllability. We adapted these two approaches to our problem setting and compared our method against them.

Local text-driven image editing. Recently, various text-driven image editing methods were proposed [1,3,6,7, 9,14,23,24,28,36,38,39,47,58,65] that allow editing an existing image. Some of the methods support *localized* image editing: GLIDE [43] and DALL·E 2 [51] train a designated inpainting model, whereas Blended Diffusion [4, 5] leverages a pretrained text-to-image model. Combining these localized methods with a text-to-image model may enable scene-based image generation. We compare our method against this approach in the supplementary.

3. Method

We aim to provide the user with more fine-grained control over the generated image. In addition to a single *global* text prompt, the user will also provide a segmentation map, where the content of each segment of interest is described using a *local* free-form text prompt.

Formally, the input consists of a global text prompt t_{global} that describes the scene in general, and a $H \times W$ raw spatio-textual matrix RST, where each entry $RST[i, j]$ contains the text description of the desired content in pixel $[i, j]$, or $\emptyset$ if the user does not wish to specify the content of this pixel in advance. Our goal is to synthesize an $H \times W$ image I that complies with both the global text description t_{global} and the raw spatio-textual scene matrix RST.

In Section 3.1 we present our novel spatio-textual representation, which we use to tackle the problem of text-to-image generation with sparse scene control. Later, in Section 3.2 we explain how to incorporate this representation into two state-of-the-art text-to-image diffusion models. Finally, in Section 3.3 we present two ways for adapting classifier-free guidance to our multi-conditional problem.

3.1. CLIP-based Spatio-Textual Representation

Over the recent years, large-scale text-to-image datasets were curated by the community, fueling the tremendous progress in this field. Nevertheless, these datasets cannot be naïvely used for our task, because they do not contain *local* text descriptions for each segment in the images. Hence, we need to develop a way to extract the objects in the image along with their textual description. To this end, we opt to use a pre-trained panoptic segmentation model [69] along with a CLIP [49] model.

CLIP was trained to embed images and text prompts into a rich shared latent space by contrastive learning on 400 million image-text pairs. We utilize this shared latent space for our task in the following way: during training we use the image encoder CLIP_{img} to extract the local embeddings using the *pixels* of the objects that we want to generate (because the local text descriptions are not available), whereas during inference we use the CLIP text encoder CLIP_{txt} to extract the local embeddings using the *text descriptions* provided by the user.

Hence, we build our spatio-textual representation, as depicted in Figure 3: for each training image x we first extract its panoptic segments $\{S_i \in [C]\}_{i=1}^{i=N}$ where C is the number of panoptic segmentation classes and N is the number of segments for the current image. Next, we randomly choose K disjoint segments $\{S_i \in [C]\}_{i=1}^{i=K}$. For each segment S_i, we crop a tight square around it, black-out the pixels in the square that are not in the segment (to avoid confusing the CLIP model with other content that might fall in the same square), resize it to the CLIP input size, and get the CLIP image embedding of that segment $\text{CLIP}_{\text{img}}(S_i)$.

Now, for the training image x we define the spatio-textual representation ST_x of shape $(H, W, \text{d}_{\text{CLIP}})$ to be:

$$ST_x[j,k] = \begin{cases} \text{CLIP}_{\text{img}}(S_i) & \text{if } [i,k] \in S_i \\ \vec{0} & \text{otherwise} \end{cases} \quad (1)$$

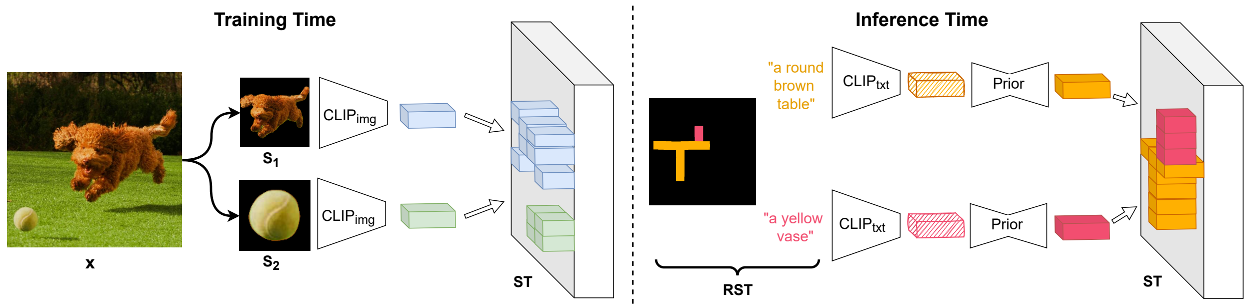

Figure 3. **Spatio-textual representation:** During training (left) — given a training image x, we extract K random segments, pre-process them and extract their CLIP image embeddings. Then we stack these embeddings in the same shapes of the segments to form the spatio-textual representation ST. During inference (right) — we embed the local prompts into the CLIP text embedding space, then convert them using the prior model P to the CLIP image embeddings space, lastly, we stack them in the same shapes of the inputs masks to form the spatio-textual representation ST.

where d_{CLIP} is the dimension of the CLIP shared latent space and $\vec{0}$ is the zero vector of dimension d_{CLIP}.

During inference time, to form the raw spatio-textual matrix RST, we embed the local prompts using CLIP_{txt} to the CLIP text embedding space. Next, in order to mitigate the domain gap between train and inference times, we convert these embeddings to CLIP_{img} using a designated prior model P. The prior model P was trained separately to convert CLIP text embeddings to CLIP image embeddings using an image-text paired dataset following DALL·E 2. Finally, as depicted in Figure 3 (right), we construct the spatio-textual representation ST using these embeddings at pixels indicated by the user-supplied spatial map. For more implementation details, please refer to the supplementary.

3.2. Incorporating Spatio-Textual Representation into SoTA Diffusion Models

The current diffusion-based SoTA text-to-image models are DALL·E 2 [51], Imagen [59] and Stable Diffusion [56]. At the time of writing this paper, DALL·E 2 [51] model architecture and weights are unavailable, hence we start by reimplementing DALL·E 2-like text-to-image model that consists of three diffusion-based models: (1) a *prior model* P trained to translate the tuples $(\text{CLIP}_{\text{txt}}(y), \text{BytePairEncoding}(y))$ into $\text{CLIP}_{\text{img}}(x)$ where (x, y) is an image-text pair, (2) a *decoder model* D that translates $\text{CLIP}_{\text{img}}(x)$ into a low-resolution version of the image $x_{64\times64}$, and (3) a *super-resolution model* SR that upsamples $x_{64\times64}$ into a higher resolution of $x_{256\times256}$. Concatenating the above three models yields a text-to-image model $SR \circ D \circ P$.

Now, in order to utilize the vast knowledge it has gathered during the training process, we opt to fine-tune a pre-trained text-to-image model in order to enable localized textual scene control by adapting its decoder component D. At each diffusion step, the decoder performs a single denoising step $x_t = D(x_{t-1}, \text{CLIP}_{\text{img}}(x), t)$ to get a less noisy version of x_{t-1}. In order to keep the spatial correspondence between the spatio-textual representation ST and the noisy image x_t at each stage, we choose to concatenate x_t and ST along the RGB channels dimensions, to get a total input of shape $(H, W, 3 + \text{d}_{\text{CLIP}})$. Now, we extend each kernel of the first convolutional layer from shape (C_{in}, K_H, K_W) to $(C_{in} + \text{d}_{\text{CLIP}}, K_H, K_W)$ by concatenating a tensor of dimension d_{CLIP} that we initialize with He initialization [27]. Next, we fine-tuned the decoder using the standard simple loss variant of Ho *et al.* [32] $L_{\text{simple}} = E_{t,x_0,\epsilon}\left[||\epsilon - \epsilon_\theta(x_t, \text{CLIP}_{\text{img}}(x_0), ST, t)||^2\right]$ where ϵ_θ is a UNet [42] model that predicts the added noise at each time step t, x_t is the noisy image at time step t and ST is our spatio-textual representation. To this loss, we added the standard variational lower bound (VLB) loss [44].

Next, we move to handle the second family of SoTA diffusion-based text-to-image models: latent-based models. More specifically, we opt to adapt Stable Diffusion [56], a recent open-source text-to-image model. This model consist of two parts: (1) an autoencoder $(Enc(x), Dec(z))$ that embeds the image x into a lower-dimensional latent space z, and, (2) a diffusion model A that performs the following denoising steps on the latent space $z_{t-1} = A(z_t, \text{CLIP}_{\text{txt}}(t))$. The final denoised latent is fed to the decoder to get the final prediction $Dec(z_0)$.

We leverage the fact that the autoencoder $(Enc(x), Dec(z))$ is fully-convolutional, hence, the latent space z *corresponds spatially* to the generated image x, which means that we can concatenate the spatio-textual representation ST the same way we did on the pixel-based model: concatenate the noisy latent z_t and ST along the channels dimensions, to get a total input of shape $(H, W, dim(z_t) + \text{d}_{\text{CLIP}})$ where $dim(z_t)$ is the number of feature channels. We initialize the newly-added channels in the kernels of the first convolutional layer using the same method we utilized for the pixel-based variant. Next, we fine-tune the denosing model

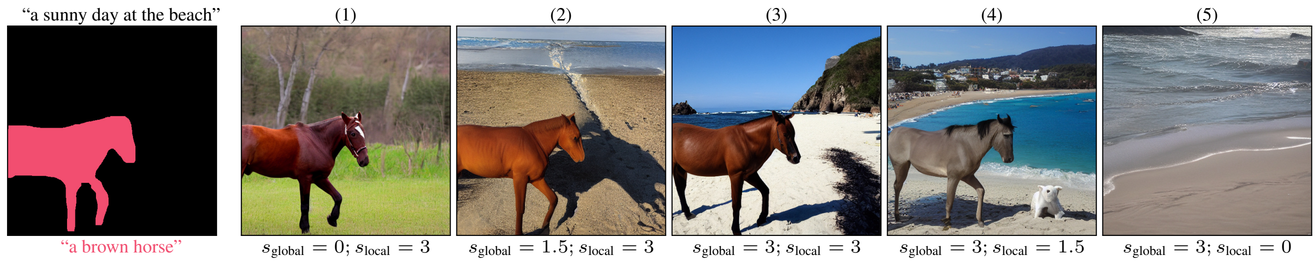

Figure 4. **Multi-scale control:** Using the multi-scale inference (Equation (3)) allows fine-grained control over the input conditions. Given the same inputs (left), we can use different scales for each condition. In this example, if we put all the weight on the local scene (1), the generated image contains a horse with the correct color and posture, but not at the beach. Conversely, if we place all the weight on the global text (5), we get an image of a beach with no horse in it. The in-between results correspond to a mix of conditions — in (4) we get a gray donkey, in (2) the beach contains no water, and in (3) we get a brown horse at the beach on a sunny day.

by $L_{\text{LDM}} = E_{t,y,z_0,\epsilon}\left[||\epsilon - \epsilon_\theta(z_t, \text{CLIP}_{\text{txt}}(y), ST, t)||^2\right]$ where z_t is the noisy latent code at time step t and y is the corresponding text prompt. For more implementation details of both models, please read the supplementary.

3.3. Multi-Conditional Classifier-Free Guidance

Classifier-free guidance [33] is an inference method for conditional diffusion models which enables trading-off mode coverage and sample fidelity. It involves training a conditional and unconditional models simultaneously, and combining their predictions during inference. Formally, given a conditional diffusion model $\epsilon_\theta(x_t|c)$ where c is the condition (e.g., a class label or a text prompt) and x_t is the noisy sample, the condition c is replaced by the null condition $\emptyset$ with a fixed probability during training. Then, during inference, we extrapolate towards the direction of the condition $\epsilon_\theta(x_t|c)$ and away from $\epsilon_\theta(x_t|\emptyset)$:

$$\hat{\epsilon}_\theta(x_t|c) = \epsilon_\theta(x_t|\emptyset) + s \cdot (\epsilon_\theta(x_t|c) - \epsilon_\theta(x_t|\emptyset)) \quad (2)$$

where $s \geq 1$ is the guidance scale.

In order to adapt classifier-free guidance to our setting, we need to extend it to support multiple conditions. Given a conditional diffusion model $\epsilon_\theta(x_t|\{c_i\}_{i=1}^{i=N})$ where $\{c_i\}_{i=1}^{i=N}$ are N condition inputs, during training, we *independently* replace each condition c_i with the null condition $\emptyset$. Then, during inference, we calculate the direction of each condition $\Delta_i^t = \epsilon_\theta(x_t|c_i) - \epsilon_\theta(x_t|\emptyset)$ separately, and linearly combine them using N guidance scales s_i by extending Eq. (2):

$$\hat{\epsilon}_\theta(x_t|\{c_i\}_{i=1}^{i=N}) = \epsilon_\theta(x_t|\emptyset) + \sum_{i=1}^{i=N} s_i \Delta_i^t \quad (3)$$

Using the above formulation, we are able to control each of the conditions separately during inference, as demonstrated in Figure 4. To the best of our knowledge, we are the first to demonstrate this effect in the multi-conditional case.

The main limitation of the above formulation is that its execution time grows linearly with the number of conditions, i.e., each denoising step requires $(N+1)$ feed-forward executions: one for the null condition and N for the other conditions. As a remedy, we propose a fast variant of the multi-conditional classifier-free guidance that trades-off the fine-grained controllability of the model with the inference speed: the training regime is identical to the previous variant, but during inference, we calculate only the direction of the joint probability of all the conditions $\Delta_{\text{joint}}^t = \epsilon_\theta(x_t|\{c_i\}_{i=1}^{i=N}) - \epsilon_\theta(x_t|\emptyset)$, and extrapolate along this *single* direction:

$$\hat{\epsilon}_\theta(x_t|\{c_i\}_{i=1}^{i=N}) = \epsilon_\theta(x_t|\emptyset) + s \cdot \Delta_{\text{joint}}^t \quad (4)$$

where $s \geq 1$ is the *single* guidance scale. This formulation requires only two feed-forward executions at each denoising step, however, we can no longer control the magnitude of each direction separately.

We would like to stress that the training regime is identical for both of these formulations. Practically, it means that the user can train the model once, and only during inference decide which variant to choose, based on the preference at the moment. Through the rest of this paper, we used the fast variant with fixed $s = 3$. See the ablation study in Section 4.4 for a comparison between these variants.

In addition, we noticed that the texts in the image-text pairs dataset contain elaborate descriptions of the entire scene, whereas we aim to ease the use for the end-user and remove the need to provide an elaborate global prompt in addition to the local ones, i.e., to not require the user to repeat the same information twice. Hence, in order to reduce the domain gap between the training data and the input at inference time, we perform the following simple trick: we concatenate the local prompts to the global prompt at inference time separated by commas.

4. Experiments

For both the pixel-based and latent-based variants, we fine-tuned pre-trained text-to-image models with 35M image-text pairs, following Make-A-Scene [22], while filtering out image-text pairs containing people.

In Section 4.1 we compare our method against the baselines both qualitatively and quantitatively. Next, in Sec-

	Automatic Metrics				User Study			Performance
Method	Global ↓ distance	Local ↓ distance	Local ↑ IOU	FID ↓	Visual quality	Global match	Local match	Inference ↓ time (sec)
NTLB [46]	0.7547	0.7814	0.1914	36.004	91.4%	85.54%	79.29%	326
MAS [22]	0.7591	0.7835	**0.2984**	21.367	81.25%	70.61%	57.81%	76
SpaText (pixel)	0.7661	0.7862	0.2029	23.128	87.11%	80.96%	71.09%	52
SpaText (latent)	**0.7436**	**0.7795**	0.2842	**6.7721**	-	-	-	**7**

Table 1. **Metrics comparison:** We evaluated our method against the baselines using automatic metrics (left) and human ratings (middle). The results for the human ratings (middle) are reported as the percentage of the majority vote raters that preferred our latent-based variant of our method over the baseline, i.e., any value above 50% means our method was favored. The inference time reported (right) are for a single image.

tion 4.2 we describe the user study we conducted. Then, in Section 4.3 we discuss the sensitivity of our method to details in the mask. Finally, in Section 4.4 we report the ablation study results.

4.1. Quantitative & Qualitative Comparison

We compare our method against the following baselines: (1) No Token Left Behind (NTLB) [46] proposes a method that conditions a text-to-image model on spatial locations using an optimization approach. We adapt their method to our problem setting as follows: the global text prompt t_{global} is converted to a full mask (that contains all the pixels in the image), and the raw spatio-textual matrix RST is converted to separate masks. (2) Make-A-Scene (MAS) [22] proposes a method that conditions a text-to-image model on a global text t_{global} and a *dense* segmentation map with *fixed labels*. We adapt MAS to support *sparse* segmentation maps of *general local prompts* by concatenating the local texts of the raw spatio-textual matrix RST into the global text prompt t_{global} as described in Section 3.3 and provide a label for each segment (if there is no exact label in the fixed list, the user should provide the closest one). Instead of a dense segmentation map, we provided a sparse segmentation map, where the background pixels are marked with an "unassigned" label.

In order to evaluate our method effectively, we need an automatic way to generate a large number of coherent inputs (global prompts t_{global} as well as a raw spatio-textual matrix RST) for comparison. Naïvely taking random inputs is undesirable, because such inputs will typically not represent a meaningful scene and may be impossible to generate. Instead, we propose to derive random inputs from *real images*, thus guaranteeing that there is in fact a possible natural image corresponding to each input. We use 30,000 samples from COCO [41] validation set that contain global text captions as well as a dense segmentation map for each sample. We convert the segmentation map labels by simply providing the text "a {label}" for each segment. Then, we randomly choose a subset of those segments to form the sparse input. Notice that for MAS, we additionally provided the ground-truth label for each segment. For more details and generated input examples, see the supplementary document.

In order to evaluate the performance of our method numerically we propose to use the following metrics that test different aspects of the model: (1) *FID score* [29] to assess the overall quality of the results, (2) *Global distance* to assess how well the model's results comply with the global text prompt t_{global} — we use CLIP to calculate the cosine distance between $\text{CLIP}_{\text{txt}}(t_{\text{global}})$ and $\text{CLIP}_{\text{img}}(I)$, (3) *Local distance* to assess the compliance between the result and the raw spatio-textual matrix RST — again, using CLIP for each of the segments in RST separately, by cropping a tight area around each segment c, feeding it to CLIP_{img} and calculating the cosine distance with $\text{CLIP}_{\text{txt}}(t_{\text{local}})$, (4) *Local IOU* to assess the shape matching between the raw spatio-textual matrix RST and the generated image — for each segment in RST, we calculate the IOU between it and the segmentation prediction of a pre-trained segmentation model [70]. As we can see in Table 1(left) our latent-based model outperforms the baselines in all the metrics, except the local IOU, which is better in MAS because our method is somewhat insensitive to the given mask shape (Section 4.3) — we view this as an advantage. In addition, we can see that our latent-based variant outperforms the pixel-based variant in all the metrics, which may be caused by insufficient reimplementation of the DALL·E 2 model. Nevertheless, we noticed that this pixel-based model is also able to take into account both the global text and spatio-textual representation. The rightmost column in Table 1 reports the inference times for a single image across the different models computed on a single V100 NVIDIA GPU. The results indicate that our method (especially the latent-based one) outperforms the baselines significantly. For more details, please read the supplementary.

In addition, Figure 5 shows a qualitative comparison between the two variants of our method and the baselines. For MAS, we manually choose the closest label from the fixed labels set. As we can see, the SpaText (latent) outperforms the baselines in terms of compliance with both the global and local prompts, and in overall image quality.

4.2. User Study

In addition, we conducted a user study on Amazon Mechanical Turk (AMT) [2] to assess the visual quality, as well as compliance with global and local prompts. For each task, the raters were asked to choose between two images generated by different models along the following dimensions: (1) overall image quality, (2) text-matching to the global prompt t_{global} and (3) text-matching to the local prompts of the raw spatio-textual matrix RST. For more details, please read the supplementary.

In Table 1 (middle) we present the evaluation results against the baselines, as the percentage of majority rates that

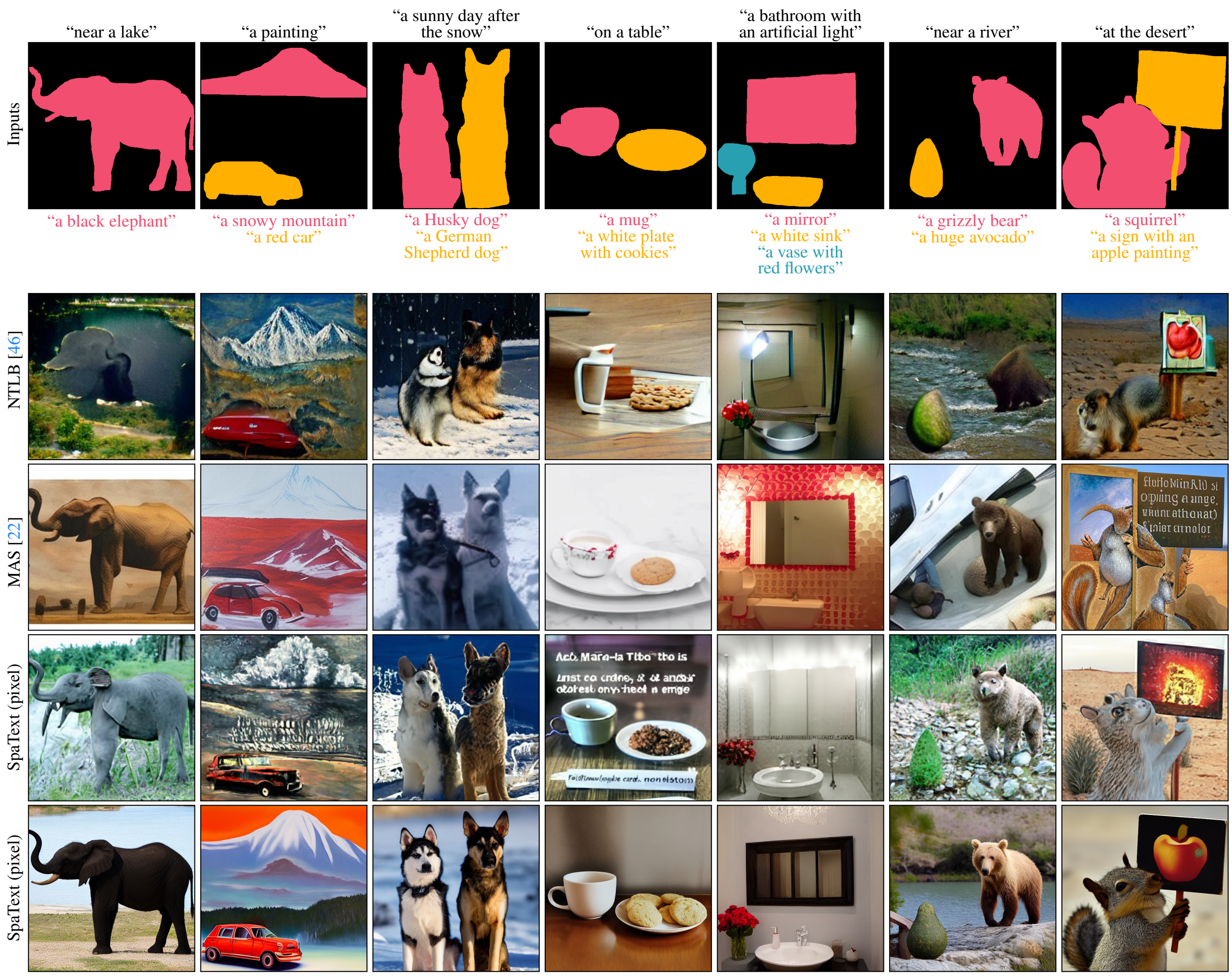

Figure 5. **Qualitative comparison:** Given the inputs (top row), we generate images using the baselines (adapted to our task as described in Section 4.1) and the two variants of our method. As we can see, SpaText (latent) outperforms the baselines in terms of compliance with both the global and local texts, and in overall image quality.

preferred our method (based on the latent model) over the baseline. As we can see, our method is preferred by human evaluators in all these aspects vs. all the baselines. In addition, NTLB [46] achieved overoptimistic scores in the CLIP-based automatic metrics — it achieved lower human evaluation ratings than Make-A-Scene [22] in the global and local text-matching aspects, even though it got better scores in the corresponding automatic metrics. This might be because NTLB is an optimization-based solution that uses CLIP for generation, hence is susceptible to adversarial attacks [26, 64].

4.3. Mask Sensitivity

During our experiments, we noticed that the model generates images that correspond to the implicit masks in the spatio-textual representation ST, but not *perfectly*. This is also evident in the local IOU scores in Table 1. Nevertheless, we argue that this characteristic can be beneficial, especially when the input mask is not realistic. As demonstrated in Figure 6, given an inaccurate, hand drawn, general animal shape (left) the model is able to generate different objects guided by the local text prompt, even when it does not perfectly match the mask. For example, the model is able to add ears (in the cat and dog examples) and horns (in the goat example), which are not presented in the input mask, or to change the body type (as in the tortoise example). However, all the results share the same pose as the input mask. One reason for this behavior might be the downsampling of the input mask, so during training some fine-grained details are lost, hence the model is incentivized to fill in the missing gaps according to the prompts.

Figure 6. **Mask insensitivity:** We found that the model is relatively insensitive to errors in the input mask. Given a general animal shape mask (left), the model is able to generate a diverse set of results driven by the different local prompts. It can add ears/horns, as in the cat, dog and goat examples or change the body type, as in the tortoise example. However, all the results share the same pose as the input mask.

	Automatic Metrics				User Study		
Scenario	Global ↓ distance	Local ↓ distance	Local ↑ IOU	FID ↓	Visual quality	Global text-match	Local text-match
(1) Binary	0.7457	0.7797	0.2973	7.6085	53.13%	50.3%	54.98%
(2) CLIP_{txt}	0.7447	0.7795	**0.3092**	7.025	58.6%	56.74%	48.53%
(3) Multiscale	0.7566	**0.7794**	0.2767	10.5896	53.61%	58.59%	55.57%
SpaText (latent)	**0.7436**	0.7795	0.2842	**6.7721**	-	-	-

Table 2. **Ablation study:** The baseline method that we used in this paper achieves better FID score and visual quality than the ablated cases. It is outperformed in terms of local IOU in (1) and (2), and in terms of local text-match in (2). The results for the human ratings (right) are reported as the percentage of the majority vote raters that preferred our SpaText (latent).

4.4. Ablation Study

We conducted an ablation study for the following cases: (1) *Binary representation* — in Section 3.1 we used the CLIP model for the spatio-textual representation ST. Alternatively, we could use a simpler binary representation that converts the raw spatio-textual matrix RST into a binary mask B of shape (H, W) by:

$$B[j,k] = \begin{cases} 1 & \text{if } RST[j,k] \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

and concatenate the local text prompts into the global prompt. (2) *CLIP text embedding* — as described in Section 3.2, we mitigate the domain gap between CLIP_{img} and CLIP_{txt} we employing a prior model P. Alternatively, we could use the CLIP_{txt} directly by removing the prior model. (3) *Multi-scale inference* — as described in Section 3.3 we used the single scale variant (Equation (4)). Alternatively, we could use the multi-scale variant (Equation (3)).

As can be seen in Table 2 our method outperforms the ablated cases in terms of FID score, human visual quality and human global text-match. When compared to the simple representation (1) our method is able to achieve better local text-match determined by the user study but smaller local IOU, one possible reason is that it is easier for the model to fit the shape of a simple mask (as in the binary case), but associating the relevant portion of the global text prompt to the corresponding segment is harder. When compared to the version with CLIP text embedding (2) our model achieves slightly less local IOU score and human local text-match while achieving better FID and overall visual quality. Lastly, the single scale manages to achieve better results than the multi-scale one (3) while only slightly less in the local CLIP distance.

5. Limitations and Conclusions

Figure 7. **Limitations:** In some cases, characteristics propagate to adjacent segments, e.g. (left), instead of a blue bowl the model generated a vase with a wooden color. In addition, the model tends to ignore tiny masks (right).

We found that in some cases, especially when there are more than a few segments, the model might miss some of the segments or propagate their characteristics. For example, instead of generating a blue bowl in Figure 7(left), the model generates a beige vase, matching the appearance of the table. Improving the accuracy of the model in these cases is an appealing research direction.

In addition, the model struggles to handle tiny segments. For example, as demonstrated in Figure 7(right), the model ignores the golden coin masks altogether. This might be caused by our fine-tuning procedure: when we fine-tune the model, we choose a random number of segments that are above a size threshold because CLIP embeddings are meaningless for very low-resolution images. For additional examples, please read the supplementary.

In conclusion, in this paper we addressed the scenario of text-to-image generation with sparse scene control. We believe that our method has the potential to accelerate the democratization of content creation by enabling greater control over the content generation process, supporting professional artists and novices alike.

References

[1] Johannes Ackermann and Minjun Li. High-resolution image editing via multi-stage blended diffusion. *arXiv preprint arXiv:2210.12965*, 2022. 3

[2] Amazon. Amazon mechanical turk. https://www.mturk.com/, 2022. 6

[3] Oron Ashual, Shelly Sheynin, Adam Polyak, Uriel Singer, Oran Gafni, Eliya Nachmani, and Yaniv Taigman. KNN-Diffusion: Image generation via large-scale retrieval. *arXiv preprint arXiv:2204.02849*, 2022. 3

[4] Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *arXiv preprint arXiv:2206.02779*, 2022. 3

[5] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18208–18218, 2022. 3

[6] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2LIVE: Text-driven layered image and video editing. *arXiv preprint arXiv:2204.02491*, 2022. 3

[7] David Bau, Alex Andonian, Audrey Cui, YeonHwan Park, Ali Jahanian, Aude Oliva, and Antonio Torralba. Paint by word. *arXiv preprint arXiv:2103.10951*, 2021. 3

[8] Andreas Blattmann, Robin Rombach, Kaan Oktay, and Björn Ommer. Retrieval-augmented diffusion models. *arXiv preprint arXiv:2204.11824*, 2022. 3

[9] Hila Chefer, Sagie Benaim, Roni Paiss, and Lior Wolf. Image-based clip-guided essence transfer. *arXiv preprint arXiv:2110.12427*, 2021. 3

[10] Hila Chefer, Shir Gur, and Lior Wolf. Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 397–406, 2021. 3

[11] Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 782–791, 2021. 3

[12] Wenhu Chen, Hexiang Hu, Chitwan Saharia, and William W Cohen. Re-imagen: Retrieval-augmented text-to-image generator. *arXiv preprint arXiv:2209.14491*, 2022. 3

[13] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014. 3

[14] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022. 3

[15] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *arXiv preprint arXiv:2209.04747*, 2022. 3

[16] Katherine Crowson, Stella Biderman, Daniel Kornis, Dashiell Stander, Eric Hallahan, Louis Castricato, and Edward Raff. Vqgan-clip: Open domain image generation and editing with natural language guidance. *arXiv preprint arXiv:2204.08583*, 2022. 3

[17] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. *Advances in Neural Information Processing Systems*, 34, 2021. 3

[18] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, et al. Cogview: Mastering text-to-image generation via transformers. *Advances in Neural Information Processing Systems*, 34:19822–19835, 2021. 3

[19] Ming Ding, Wendi Zheng, Wenyi Hong, and Jie Tang. Cogview2: Faster and better text-to-image generation via hierarchical transformers. *arXiv preprint arXiv:2204.14217*, 2022. 3

[20] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proc. CVPR*, pages 12873–12883, 2021. 3

[21] Stanislav Frolov, Prateek Bansal, Jörn Hees, and Andreas Dengel. Dt2i: Dense text-to-image generation from region descriptions. *arXiv preprint arXiv:2204.02035*, 2022. 3

[22] 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. *arXiv preprint arXiv:2203.13131*, 2022. 2, 3, 5, 6, 7

[23] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 3

[24] Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022. 3

[25] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014. 3

[26] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 7

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015. 4

[28] 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. 3

[29] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 6

[30] Tobias Hinz, Stefan Heinrich, and Stefan Wermter. Generating multiple objects at spatially distinct locations. In *International Conference on Learning Representations*, 2018. 3

[31] Tobias Hinz, Stefan Heinrich, and Stefan Wermter. Semantic object accuracy for generative text-to-image synthesis. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 3

[32] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proc. NeurIPS*, 2020. 3, 4

[33] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 2, 5

[34] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 3

[35] Seunghoon Hong, Dingdong Yang, Jongwook Choi, and Honglak Lee. Inferring semantic layout for hierarchical text-to-image synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7986–7994, 2018. 3

[36] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. *arXiv preprint arXiv:2210.09276*, 2022. 3

[37] Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3

[38] Chaerin Kong, DongHyeon Jeon, Ohjoon Kwon, and Nojun Kwak. Leveraging off-the-shelf diffusion model for multi-attribute fashion image manipulation. *arXiv preprint arXiv:2210.05872*, 2022. 3

[39] Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Diffusion models already have a semantic latent space. *arXiv preprint arXiv:2210.10960*, 2022. 3

[40] Wenbo Li, Pengchuan Zhang, Lei Zhang, Qiuyuan Huang, Xiaodong He, Siwei Lyu, and Jianfeng Gao. Object-driven text-to-image synthesis via adversarial training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12174–12182, 2019. 3

[41] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 6

[42] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 4

[43] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 3

[44] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *Proc. ICML*, pages 8162–8171, 2021. 3, 4

[45] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008. 3

[46] Roni Paiss, Hila Chefer, and Lior Wolf. No token left behind: Explainability-aided image classification and generation. *arXiv preprint arXiv:2204.04908*, 2022. 3, 6, 7

[47] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2085–2094, 2021. 3

[48] Dario Pavllo, Aurelien Lucchi, and Thomas Hofmann. Controlling style and semantics in weakly-supervised image generation. In *European conference on computer vision*, pages 482–499. Springer, 2020. 3

[49] 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, 3

[50] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67, 2020. 3

[51] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. *arXiv preprint arXiv:2204.06125*, 2022. 1, 2, 3, 4

[52] 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, 3

[53] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with VQ-VAE-2. *Advances in neural information processing systems*, 32, 2019. 3

[54] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In *Proc. ICLR*, pages 1060–1069, 2016. 3

[55] Scott E Reed, Zeynep Akata, Santosh Mohan, Samuel Tenka, Bernt Schiele, and Honglak Lee. Learning what and where to draw. *Advances in neural information processing systems*, 29, 2016. 3

[56] 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. 2, 3, 4

[57] Robin Rombach, Andreas Blattmann, and Björn Ommer. Text-guided synthesis of artistic images with retrieval-augmented diffusion models. *arXiv preprint arXiv:2207.13038*, 2022. 3

[58] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022. 3

[59] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi,

Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 1, 3, 4

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

[61] Wei Sun and Tianfu Wu. Image synthesis from reconfigurable layout and style. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10531–10540, 2019. 3

[62] Wei Sun and Tianfu Wu. Learning layout and style reconfigurable gans for controllable image synthesis. *IEEE transactions on pattern analysis and machine intelligence*, 44(9):5070–5087, 2021. 3

[63] Tristan Sylvain, Pengchuan Zhang, Yoshua Bengio, R Devon Hjelm, and Shikhar Sharma. Object-centric image generation from layouts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2647–2655, 2021. 3

[64] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 7

[65] Dani Valevski, Matan Kalman, Yossi Matias, and Yaniv Leviathan. Unitune: Text-driven image editing by fine tuning an image generation model on a single image. *arXiv preprint arXiv:2210.09477*, 2022. 3

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

[67] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 3

[68] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 3

[69] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. 2019. 2, 3

[70] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34:12077–12090, 2021. 6

[71] 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. 3

[72] 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. 1, 2, 3

[73] 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 *Proc. ICCV*, pages 5907–5915, 2017. 3

[74] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. StackGAN++: Realistic image synthesis with stacked generative adversarial networks. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1947–1962, 2018. 3

[75] Bo Zhao, Lili Meng, Weidong Yin, and Leonid Sigal. Image generation from layout. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8584–8593, 2019. 3