Composite Diffusion: *whole* $>= \Sigma$*parts* (Supp - Supplementary Material)

Vikram Jamwal
TCS Research, India
vikram.jamwal@tcs.com

Ramaneshwaran S*
NVIDIA, India
s.ramaneswaran2000@gmail.com

1. Supplementary organization

In this Supplementary(Supp), we provide the supplemental material to the paper: Composite Diffusion: *whole* $>= \Sigma$*parts*. It is organized into the following four main parts:

1. **Background for methods** Supp section-2 provides the mathematical background for image generation using diffusion models relevant to this paper.
2. **Our base setup and serial inpainting method** Supp section-3 provides the details of our experimental setup, the features and details of the base implementation model, and text-to-image generation through the base model which also serves as our baseline 1. Supp section-4 provides the details of our implementation of the serial inpainting method which also serves as our baseline 2.
3. **Our method: details and features** Supp section-5 covers the additional implementation details of our Composite Diffusion method discussed in the main paper. Supp section-5.3 discusses the implication of Composite Diffusion in personalizing content generation at a scale. Supp section-5.4 discusses some of the limitations of our approach and Supp section-5.5 discusses the possible societal impact of our work.
4. **Details: Related work and Evaluation** Supp section-6 provides a more detailed comparison with the related work. Supp section-7 and section-8 cover the additional details of the surveys in the human evaluation, and automated methods for evaluation respectively. Supp section-9 provides a discussion of results for each quality parameter. Supp section-10 describes the validation exercise with an external artist.

2. Background for methods

In this section, we provide an overview of diffusion-based generative models and diffusion guidance mechanisms that serve as the foundational blocks of the methods in this paper. The reader is referred to [14, 21, 47] for any further details and mathematical derivations.

2.1. Diffusion models(DM)

In the context of image generation, DMs are a type of generative model with two diffusion processes: (i) a *forward diffusion process*, where we define a Markov chain by gradually adding a small amount of random noise to the image at each time step, and (ii)a *reverse diffusion process*, where the model learns to generate the desired image, starting from a random noise sample.

2.1.1 Forward diffusion process

Given a real distribution $q(\mathbf{x})$, we sample an image $\mathbf{x}_0$ from it ($\mathbf{x}_0 \sim q(\mathbf{x})$). We gradually add Gaussian noise to it with a variance schedule $\{\beta_t \in (0,1)\}_{t=1}^{T}$ over T steps to get progressively noisier versions of the image $\mathbf{x}_1, \ldots, \mathbf{x}_T$. The conditional distribution at each time step t with respect to its previous timestep $t-1$ is given by the diffusion kernel:

$$q(\mathbf{x}_{1:T}) = q(\mathbf{x}_0) \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}) \tag{1}$$

The features in $\mathbf{x}_0$ are gradually lost as step t becomes larger. When T is sufficiently large, $T \to \infty$, then $\mathbf{x}_T$ approximates an isotropic Gaussian distribution.

Q-sampling: An interesting property of the forward diffusion process is that we can also sample $\mathbf{x}_t$ directly from $\mathbf{x}_0$ in the closed form. If we let $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$, we get:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}) \tag{2}$$

*Work performed while working at TCS Research.

Further, for $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\mathbf{x}_t$ can be expressed as a linear combination of $\mathbf{x}_0$ and $\boldsymbol{\epsilon}$:

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon} \tag{3}$$

We utilize this property in many of our algorithms and refer to it as: *'q-sampling'*.

2.1.2 Reverse diffusion process

Here we reverse the Markovian process and, instead, we sample from $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$. By repeating this process, we should be able to recreate the true sample (image), starting from the pure noise $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. If β_t is sufficiently small, $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ too will be an isotropic Gaussian distribution. However, it is not straightforward to estimate $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ in closed form. We, therefore, train a model p_θ to approximate the conditional probabilities that are required to run the reverse diffusion process.

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t)) \tag{4}$$

where $\boldsymbol{\mu}_\theta$ and $\boldsymbol{\Sigma}_\theta$ are the predicted mean and variance of the conditional Gaussian distribution. In the earlier implementations $\boldsymbol{\Sigma}_\theta(x_t, t)$ was kept constant [21], but later it was shown that it is preferable to learn it through a neural network that interpolates between the upper and lower bounds for the fixed covariance [13].

The reverse distribution is:

$$p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_T)\prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) \tag{5}$$

Instead of directly inferring the image through $\boldsymbol{\mu}_\theta(x_t, t)$), it might be more convenient to predict the noise ($\boldsymbol{\epsilon}_\theta(x_t, t)$) added to the initial noisy sample ($\mathbf{x}_t$) to obtain the denoised sample ($\mathbf{x}_{t-1}$) [21]. Then, $\boldsymbol{\mu}_\theta(\mathbf{x}_t, t)$ can be derived as follows:

$$\boldsymbol{\mu}_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) \tag{6}$$

Sampling: Mostly, a U-Net neural architecture [41] is used to predict the denoising amount at each step. A scheduler samples the output from this model. Together with the knowledge of time step t, and the input noisy sample $\mathbf{x}_t$, it generates a denoised sample $\mathbf{x}_t$. For sampling through Denoising Diffusion Probabilistic Model (DDPM) [21], denoised sample is obtained through the following computation:

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) + \sigma_t\boldsymbol{\epsilon} \tag{7}$$

where $\boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t) = \sigma_t^2\mathbf{I}$, and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a random sample from the standard Gaussian distribution.

To achieve optimal results for image quality and speed-ups, besides DDPM, various sampling methods, such as DDIM, LDMS, PNDM, and LMSD [1, 28] can be employed.

We use DDIM (Denoising Diffusion Implicit Models) as the common method of sampling for all the algorithms discussed in this paper. Using DDIM, we sample $\mathbf{x}_{t-1}$ from $\mathbf{x}_t$ and $\mathbf{x}_0$ via the following equation [46]:

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) + \sigma_t\boldsymbol{\epsilon} \tag{8}$$

Using DDIM sampling, we can produce samples that are comparable to DDPM samples in image quality, while using only a small subset of DDPM timesteps (e.g., 50 as opposed to 1000).

2.1.3 Latent diffusion models(LDM)

We can further increase the efficiency of the generative process by running the diffusion process in latent space that is lower-dimensional than but perceptually equivalent to pixel space. Performing diffusion in lower dimensional space provides massive advantages in terms of reduced computational complexity. For this, we first downsample the images into a lower-dimensional latent space and then upsample the results from the diffusion process into the pixel space. For example, the latent diffusion model described in [40] uses a suitably trained variational autoencoder to encode an RGB pixel-space image ($\mathbf{x} \in \mathbb{R}^{H\times W\times 3}$) into a latent-space representation ($\mathbf{z} = \mathcal{E}(\mathbf{x})$, $\mathbf{z} \in \mathbb{R}^{h\times w\times c}$), where $f = H/h = W/w$ describes the downsampling factor. The diffusion model in the latent space operates similarly to the pixel-space diffusion model described in the previous sections, except that it utilizes a latent space time-conditioned U-Net architecture. The output of the diffusion process ($\tilde{\mathbf{z}}$) is decoded back to the pixel-space ($\tilde{\mathbf{x}} = \mathcal{D}(\tilde{\mathbf{z}})$).

2.2. Diffusion guidance

An unconditional diffusion model, with mean $\mu_\theta(x_t)$ and variance $\Sigma_\theta(x_t)$ usually predicts a score function $\nabla_{x_t}\log p(x_t)$ which additively perturbs it and pushes it in the direction of the gradient. In conditional models, we try to model conditional distribution $\nabla_{x_t}\log p(x_t|y)$, where y can be any conditional input such as class label and free-text. This term, however, can be derived to be a combination of unconditional and conditional terms [14]:

$$\nabla_{x_t}\log p(x_t|y) = \nabla_{x_t}\log p(x_t) + \nabla_{x_t}\log p(y|x_t)$$

2.2.1 Classifier driven guidance

We can obtain $\log p(y|x_t)$ from an external classifier that can predict a target y from a high-dimension input like an image x. A guidance scale s can further amplify the conditioning guidance.

$$\nabla_{x_t} \log p_s(x_t|y) = \nabla_{x_t} \log p(x_t) + s.\nabla_{x_t} \log p(y|x_t)$$

s affects the quality and diversity of samples.

2.2.2 CLIP driven guidance

Contrastive Language–Image Pre-training (CLIP) is a neural network that can learn visual concepts from natural language supervision [37]. The pre-trained encoders from the CLIP model can be used to obtain semantic image and text embeddings which can be used to score how closely an image and a text prompt are semantically related.

Similar to a classifier, we can use the gradient of the dot product of the image and caption encodings ($f(x_t)$ and $g(c)$) with respect to the image to guide the diffusion process [18, 30, 34].

$$\hat{\mu}_\theta(x_t|c) = \mu_\theta(x_t|c) + s \cdot \Sigma_\theta(x_t|c) \nabla_{x_t}(f(x_t) \cdot g(c))$$

To perform a simple classifier-guided diffusion, Dhariwal and Nichol [13] use a classifier that is pre-trained on noisy images to guide the image generation. However, training a CLIP model from scratch on noisy images may not be always feasible or practical. To mitigate this problem we can estimate a clean image $\hat{x_0}$ from a noisy latent x_t by using the following equation.

$$\hat{x_0} = \frac{x_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(x_t, t)}{\sqrt{\bar{\alpha}_t}} \quad (9)$$

We can then use this projected clean image $\hat{x}_0$ at each state of diffusion step t for comparing with the target text. Now, a CLIP-based loss L_{CLIP} may be defined as the cosine distance (or some similar distance measure) between the CLIP embedding of the text prompt (d) and the embedding of the estimated clean image $\hat{x}_0$:

$$L_{CLIP}(x, d) = D_c(CLIP_{img}(\hat{x}_0), CLIP_{txt}(d))$$

2.2.3 Classifier-free guidance

Classifier-guided mechanisms face a few challenges, such as: (i) may not be robust enough in dealing with noised samples in the diffusion process,(ii) not all the information in x is relevant for predicting y, which may cause adversarial guidance, (iii) do not work well for predicting complex y like 'text'. The classifier-free guidance [22] helps overcome this and also utilizes the knowledge gained by a pure generative model. A conditional generative model is trained to act as both conditional and unconditional (by dropping out the conditional signal by 10-20% during the training phase). The above equation (section 3.3.1) can be reinterpreted as [14, 30]:

$$\begin{aligned}\nabla_{x_t} \log p_s(x_t|y) = \nabla_{x_t} \log p(x_t) \\ + s.(\nabla_{x_t} \log p(x_t|y) - \nabla_{x_t} \log p(x_t)) \quad (10)\end{aligned}$$

For $s = 0$, we get an unconditional model, for $s = 1$, we get a conditional model, and for $s > 1$ we strengthen the conditioning signal. The above equation can be expressed in terms of noise estimates at diffusion timestep t, as follows:

$$\hat{\epsilon}_\theta(x_t|c) = \epsilon_\theta(x_t|\emptyset) + s \cdot (\epsilon_\theta(x_t|c) - \epsilon_\theta(x_t|\emptyset)) \quad (11)$$

where c is the text caption representing the conditional input, and $\emptyset$ is an empty sequence or a null set representing unconditional output. Our DDIM sampling for conditioned models will utilize these estimates.

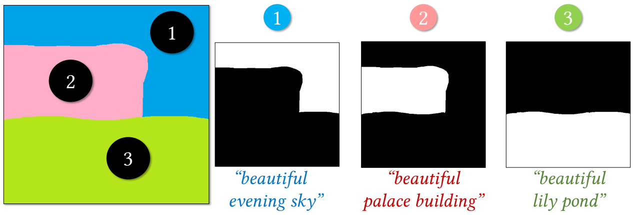


Figure 1. Running Example: Free-form segment layout and natural text input

3. Our experimental setup

As stated earlier, in this work, we aim to generate a composite image guided entirely by free-form segments and corresponding natural textual prompts (with optional additional control conditions). In this section, we summarize our choice of base setup, provide a running example to help explain the working of different algorithms, and provide implementation details of the base setup.

3.1. Running example

To explain the different algorithms, we will use a common running example. The artist's input is primarily bimodal: free-form segment layout and corresponding natural language descriptions as shown in Figure 1. As a first step common to all the algorithms, the segment layout is converted to segment masks as one-hot encoding vectors where '0' represents the absence of pixel information, and '1' indicates the presence of image pixels. To standardize the outputs of the generative process, all the initial inputs (noise samples, segment layouts, masks, reference, and background images) and the generated images in this paper are of 512x512 pixel dimensions. Additionally, in the case of latent diffusion setup, we downsize the masks, encode the reference images, and sample the noise into 64x64 pixels corresponding to the latent space dimensions of the model.

3.2. Implementation details

We choose open-domain diffusion model architecture, namely *Stable Diffusion* [40], to serve as base architectures for our composite diffusion methods. Table 1 provides a summary of the features of the base setup. The diffusion model has a U-Net backbone with a cross-attention mechanism, trained to support conditional diffusion. We use the pre-trained text-to-image diffusion model (Version 1.5) that is developed by researchers and engineers from CompVis, Stability AI, RunwayML, and LAION and is trained on 512x512 images from a subset of the LAION-5B dataset. A frozen CLIP ViT-L/14 text encoder is used to condition the model on text prompts. For scheduling the diffusion steps and sampling the outputs, we use DDIM [46].

Table 1. Summary of features of the base setup

Feature	Setup
Diffusion Space	Latent
Conditionality	Conditional
Guidance	Classifier-free
Model Size	≈ 850 million
Open Domain Models	StabilityAI
Sampling Method	DDIM

Algorithm 1: Text-to-Image generation in the base setup

```
Input Target text description $d$,
Initial image, $x_T \sim \mathcal{N}(0, \mathbf{I})$, Number of diffusion
  steps $= k$.
Output: An output image, $x_0$, which is sufficiently
  grounded to input $d$.
$z_T \leftarrow \mathcal{E}(x_T)$, ;                ◁ Encode into latent space
$d_z \leftarrow \mathcal{C}(d)$ ;                   ◁ Create CLIP text encoding
for all $t$ from $k$ to 1 do
    $z_{t-1} \leftarrow Denoise(z_t, d_z)$ ;        ◁ Denoise using
      text-condition and DDIM
end
return $x_0 \leftarrow \mathcal{D}(z_0)$ ;          ◁ Final Image
```

We describe image generation through this setup in the next section.

3.3. Text-to-Image generation in the base setup

In this setup (refer to Figure 2), a pixel-level image (x) is first encoded into a lower-dimensional latent-space representation with the help of a variational autoencoder(VAE) $(\mathcal{E}(x) \rightarrow z)$. The diffusion process then operates in this latent space. This setup uses a conditional[1] diffusion model which is pre-trained on natural text using CLIP encoding. For a generation, the model takes CLIP encoding of the natural text $(\mathcal{C}(d) \rightarrow d_{CLIP})$ as the conditioning input and directly infers a denoised sample z_t without the help of an external classifier (classifier free guidance) [22]. Mathematically, we use equation 11 for generating the additive noise $\hat{\epsilon}_\theta$ at timestep t, and use equation 8 for generating z_t from $\hat{\epsilon}_\theta$

[1] In practice, the model is trained to act as a both conditional and unconditional model. An empty text prompt is used for unconditional generation along with the input text prompt for conditional generation. The two results are then combined to generate a better quality denoised image. Refer to section 2.2.3.

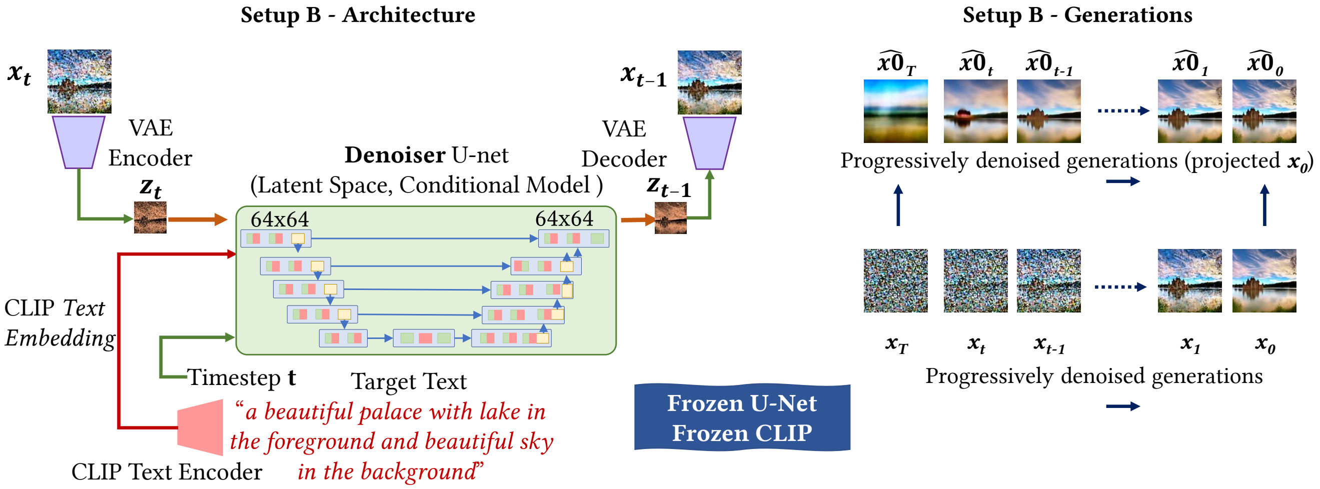


Figure 2. Base setup generation with latent-space diffusion and classifier-free implicit guidance

via DDIM sampling. After the diffusion process is over, the resultant latent z_0 is decoded back to pixel-space ($\mathcal{D}(z_0) \rightarrow x_0$).

As stated earlier, spatial information cannot be adequately described through only text conditioning. In the next section, we extend the existing in-painting methods to support Composite Diffusion. However, we shall see that these methods do not fully satisfy our quality desiderata which leads us to the development of our approach for Composite Diffusion as described in the main paper.

4. Composite Diffusion through serial inpainting

Inpainting is the means of filling in missing portions or restoring the damaged parts of an image. It has been traditionally used to restore damaged photographs and paintings and (or) to edit and replace certain parts or objects in digital images [8]. Diffusion models have been quite effective in inpainting tasks. A portion of the image, that needs to be edited, is marked out with the help of a mask, and then the content of the masked portion is generated through a diffusion model - in the context of the rest of the image, and sometimes with the additional help of a text prompt [3, 5, 27].

An obvious question is: Can we serially (or repeatedly) apply inpainting to achieve Composite Diffusion? In the following section, we develop our implementation for serial inpainting and discuss issues that arise with respect to Composite Diffusion achieved through these means. The implementation also serves as the baseline for comparing our main Composite Diffusion algorithms.

4.1. Serial Inpainting - algorithm and implementation

The method essentially involves successive application of the in-painting method for each segment of the layout. We start with an initial background image (I_{bg}) and repeatedly apply the in-painting process to generate segments specified in the free-form segment layout and text descriptions (refer to Algo. 2 for details). The method is further explained in Fig. 3

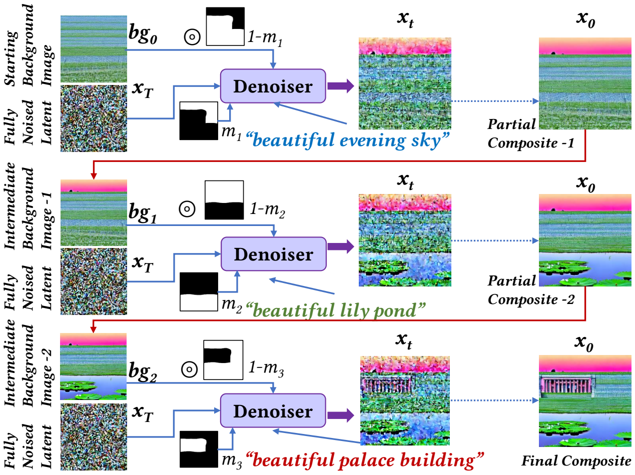


Figure 3. Diffusion steps in the algorithm for Serial Inpainting. Starting with an initial background image bg_0, we inpaint a segment into it to get x_0. The new image x_0 serves as the background image for the next stage inpainting process to generate the new x_0 with the inpainted second segment. The process is repeated till we have inpainted all the segments. The final x_0 is the generated *composite*.

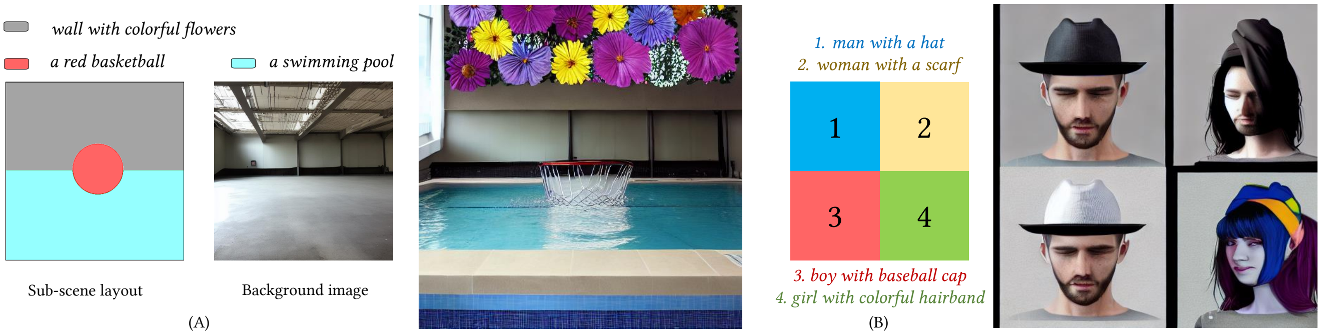


Figure 4. Some of the issues in serial-inpainting: (A) The background image plays a dominant part in the composition, and sometimes the prompt specifications are missed if the segment text-prompt does not fit well into the background image context, e.g., missing red basketball in the swimming pool, (B) The earlier stages of serial-inpainting influence the later stages; in this case, the initial background image is monochrome black, the first segment is correctly generated but in the later segment generations, the segment-specific text-prompts are missed and duplicates are created.

with the help of the running example.

Algorithm 2: Serial Inpainting for composite creation

```
Input: Set of segment masks $m^i \in M$, set of segment descriptions $d^i \in D$, background image $I_{bg}$, initial image, $x_T \sim \mathcal{N}(0, \mathbf{I})$
Output: An output image, $x_{comp}$, which is sufficiently grounded to the inputs of segment layout and segment descriptions.
$z_T \leftarrow \mathcal{E}(x_T)$, ;   ◁ Encode into latent space
$\forall i, m_z^i \leftarrow Downsample(m^i)$ ;   ◁ Downsample all masks to latent space
$\forall i, d_z^i \leftarrow \mathcal{C}_{CLIP}(d^i)$ ;   ◁ Generate CLIP encoding for all text descriptions
for all segments $i$ from 1 to $n$ do
    $z_{bg}^{masked} \leftarrow \mathcal{E}(I_{bg} \odot (1 - m^i))$ ;   ◁ Encode masked background image
    $z_{bg} \leftarrow Inpaint(z_T, z_{bg}^{masked}, m_z^i, d_z^i)$;   ◁ Inpaint the segment
    $I_{bg} \leftarrow \mathcal{D}(z_{bg})$ ;   ◁ Decode the latent to get the new reference image
end
return $x^{comp} \leftarrow I_{bg}$ ;   ◁ Final composite
```

We base our implementation upon the specialized in-painting method developed by RunwayML for Stable Diffusion [40]. This in-painting method extends the U-net architecture described in the previous section to include additional input of a masked image. It has 5 additional input channels (4 for the encoded masked image and 1 for the mask itself) and a checkpoint model which is fine-tuned for in-painting.

4.2. Issues in Composite Diffusion via serial inpainting

The method is capable of building good composite images. However, there are a few issues. One of the main issues with the serial inpainting methods for Composite Diffusion is the *dependence on an initial background image*. Since this method is based on inpainting, the segment formation cannot start from scratch. So a suitable background image has to be either picked from a collection or generated anew. If we generate it anew, there is no guarantee that the segments will get the proper context for development. This calls for a careful selection from multiple generations. Also because a new segment will be generated in the context of the underlying image, this sometimes leads to undesirable consequences. Further, if any noise artifacts or other technical aberrations get introduced in the early part of the generation, their effect might get amplified in the repeated inpainting process. Some other issues might arise because of a specific inpainting implementation. For example, in the method of inpainting that we used (RunwayML Inpainting 1.5), the mask text inputs were occasionally missed and sometimes the content of the segments was duplicated. Refer to Fig. 4 for visual examples of some of these issues.

All these issues motivated the need to develop our methods, as described in the main paper, to support Composite Diffusion. We compare our algorithms against these two baselines of (i) basic text-to-image algorithms, and (ii) serial inpainting algorithms. The results of these comparisons are presented in the main paper with some more details available in the later sections of this Supplementary.

5. Our method: details and features

In the main paper, we presented a generic algorithm that is applicable to any diffusion model that supports *conditional generation with classifier-free implicit guidance*. Here, we present the implementation details and elaborate on a few downstream applications of Composite Diffusion.

5.1. Implementation details of the main algorithm

In the previous Supplementary section 3, we detailed the actual base model which we use as the example implementation of Composite Diffusion. Since the base setup operates in latent diffusion space, to implement our main Composite Diffusion algorithm in this setup, we have to do two additional steps: **(i)** Prepare the input for latent diffusion by decoding all the image latents through a VAE to 64x64 latent pixel space, **(ii)** After the Composite Diffusion process (refer to Fig. 6 for the details of typical steps), use a VAE decoder to decode the outputs of the latent diffusion model into the 512x512 pixel space. Since the VAE encoding maintains the spatial information, we either directly use a 64x64 pixel segment layout, or downsize the resulting masks to 64x64 pixel image space.

As mentioned in the main paper, for supporting additional control conditions in Composite Diffusion, we use the Stable Diffusion v1.5 compatible implementation of ControlNet [49]. ControlNet is implemented as a parallel U-Net whose weights are copied from the main architecture, but which can be trained on particular control conditions [49] such as canny edge, lineart, scribbles, semantic segmentations, and open poses.

In our implementation, for supporting *control conditions* in segments, we first prepare a control input for every segment. The controls that we experimented with included lineart, open-pose, and scribble. Each segment has a separate control input that is designed to be formed in a 512x512 image space but only in the region that is specific to that segment. Each control input is then passed through an encoding processor that creates a control condition that is embedded along with the text conditioning. ControlNets convert image-based conditions to 64 × 64 feature space to match the convolution size: $c_f = \mathcal{E}(c_i)$ (refer to equation 9 of [49]), where c_i is the image space condition, and c_f is the corresponding converted feature map.

Another important aspect is to use a ControlNet model that is particularly trained for the type of control input specified for a segment. However, as shown in the main paper and also illustrated in Fig. 1 in the main paper, more than one type of ControlNets can be deployed for different segments to achieve Composite Diffusion.

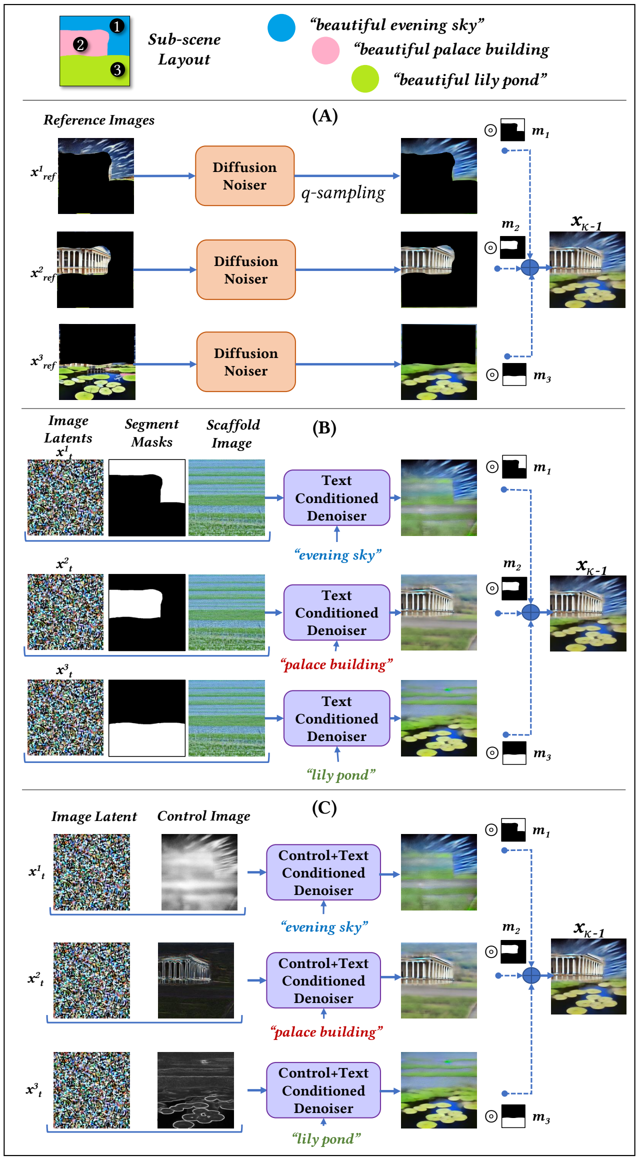


Figure 5. Scaffolding stage step for three different cases: (A) with reference images, (B) with a scaffolding image, and (C) with control conditions. Please note that for case (A), the *diffusion noising* process is only a single step, while for cases (B) and (C), the *diffusion denoising* process repeats for each time step till the end of scaffolding stage at $t = \kappa$. All the segments develop independently of each other. The individual segments are composed to form an intermediate composite only at the end of the scaffolding stage.

5.2. Example runs

With reference to the running example shown in the main paper, we present the different stages of the evolution of a composite image using Serial Inpainting and our Composite Diffusion algorithms. Refer to Figures 12, 13, 14, 15, and 16. To standardize our

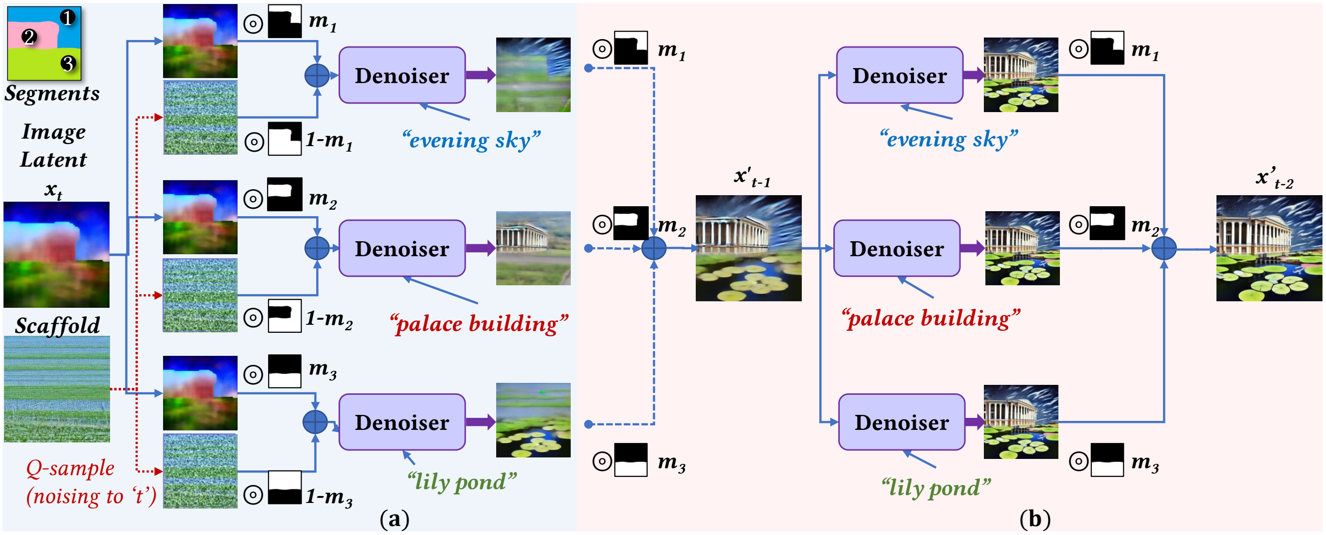


Figure 6. Typical diffusion steps in the two-stage Composite Diffusion process using a scaffolding image: **(a)** During the scaffolding stage, each segment is generated in *isolation* using a separate diffusion process after composing with the noised scaffolding image. **(b)** During the harmonizing stage, the final composed latent from the scaffolding stage is iteratively denoised using separate diffusion processes with segment-specific conditioning information; the segments are *composed* after every diffusion timestep for harmonization.

depiction, we run each algorithm for a total of 50 diffusion steps using DDIM as the underlying sampling method. The figures show every alternate DDIM step.

The scaffolding images used in the text-only case of Composite Diffusion, do influence the characteristics of the generated image. Though we allow the use of any arbitrary scaffolding image, an artist is advised to make judicious use of scaffolding images suitable for her particular artwork. Fig. 9 visually illustrates the impact of the scaffolding image on the generated image with the help of a few scaffolding examples.

5.3. Personalization at a scale

One of the motivations for composite image generation is to produce a controlled variety of outputs. This is to enable customization and personalization at a scale. Our Composite Diffusion models help to achieve variations through: (i) variation in the initial noise sample, (ii) variation in free-form segment layout, (iii) variation through segment content, and (iv) variation through fine-tuned models.

5.3.1 Variation through Noise

This is applicable to all the generative diffusion models. The initial noise sample massively influences the final generated image. This initial noise can be supplied by a purely random sample of noise or by an appropriately noised (*q-sampled*) reference image. Composite Diffusion further allows us to keep these initial noise variations particular to a segment. This method, however, only gives more variety but not any control over the composite image generations.

5.3.2 Variation through segment layout

We can introduce controlled variation in the spatial arrangement of elements or regions of an image by changing the segment layout while keeping the segment descriptions constant. Refer to figure 10 for an illustration where we introduce two different layouts for any given set of segment descriptions.

5.3.3 Variation through text descriptions

Alternatively, we can keep the segment layout constant, and change the description of the segments (through text or control conditions) to bring controlled variation in the content of the segments. Refer to figure 10 for an illustration where each of the three columns represents a different set of segment descriptions for any of the segment layouts.

5.3.4 Specialized fine-tuned models

The base diffusion models can be further fine-tuned on specialized data sets to produce domain-specialized image generations. For example, a number of fine-tuned implementations of Stable Diffusion are available in the public domain [2]. This aspect can be extremely useful

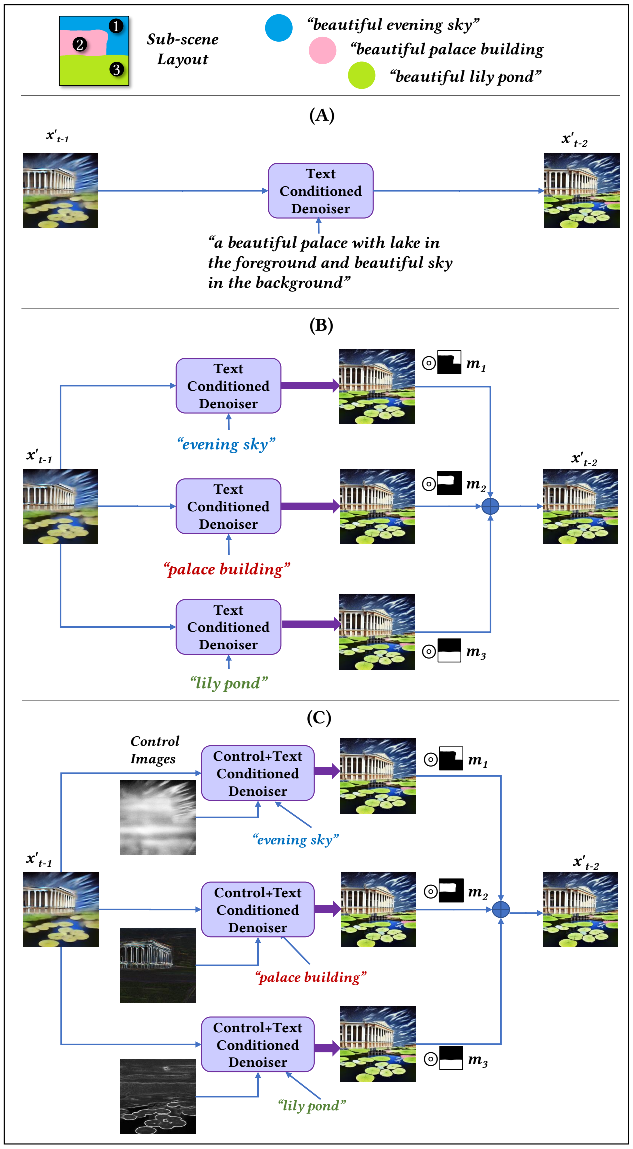


Figure 7. Harmonization stage step for three different cases: (A) a single global text description, (B) sub-scene specific text description, and (C) sub-scene specific text description and control condition. Please note that for all the cases, the harmonization stage starts with the output of the scaffolding stage composite latent. For case (A), there is no composition step, while for cases (B) and (C), the composition step follows the denoising steps for every timestep.

when creating artwork customized for different sets of consumers. One of the advantages of our composite methods is that as long as the fine-tuning does not disturb the base-model architecture, *our methods allow a direct plug-and-play with the fine-tuned models.*

Figure 11 gives an illustration of using 10 different public domain fine-tuned models with our main Composite Diffusion algorithm for generating specific-styled artwork. The only code change required for achieving these results was the change of reference to the fine-tuned model and the addition of style specification in the text prompts.

In the following sections, we discuss some of the limitations of our approach and provide a brief discussion on the possible societal impact of this work.

5.4. Limitations

Though our method is very flexible and effective in a variety of domains and composition scenarios, we do encounter some limitations which we discuss below:

Granularity of sub-scenes: The segment sizes in the segment layout are limited by the diffusion image space. So, as the size of the segment grows smaller, it becomes difficult to support sub-scenes. Our experience has shown that it is best to restrict the segment layout to 2-5 sub-scenes. Some of this is due to the particular model that we use in implementation. Since Stable Diffusion is a latent space diffusion model [40], the effective size for segment layout is only 64x64 pixels. If we were operating directly in the pixel space, we would have considerably more flexibility because of 8 fold increase in the segment-layout size of 512x512 pixels.

Shape conformance: In the only text-only conditioning case, our algorithms do perform quite well on mask shape conformance. However, total shape adherence to an object only through the segment layout is sometimes difficult. Moreover, in the text-only condition case, while generating an image within a segment the whole latent is in play. The effectiveness of a generation within the segment is influenced by how well the scaffolding image is conducive as well as non-interfering to the image requirements of the segment. This creates some dependency on the choice of scaffolding image. Further, extending the scaffolding stage improves the conformance of objects to mask shapes but there is a trade-off with the overall harmony of the image.

So in the case where strict object conformance is required, we recommend using the control condition inputs as specified in our algorithm, though this might reduce the diversity of the images that text-only conditioning can produce.

*Training and model limitations:*The quality and variety in generated object configurations are heavily influenced by the variety that the model encounters in the training data. So, as a result, not all object specifications are equal in creating quality artifacts. Although we have tested the models and methods on different kinds of compositions, based on our limited usage we cannot claim that model will equally work well for all domains. For example, we find that it works

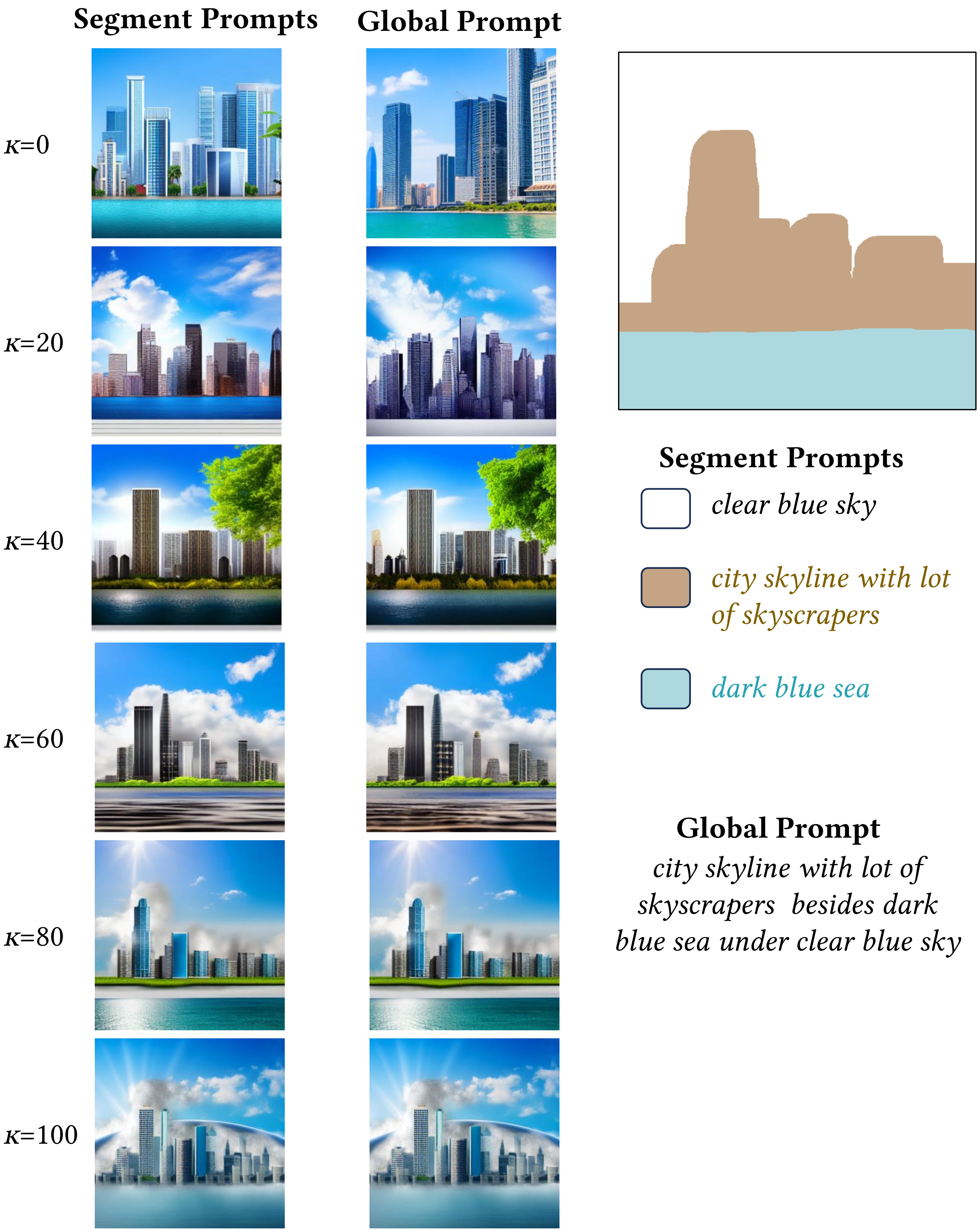

Segment Prompts
Global Prompt
κ=0
κ=20
κ=40
κ=60
κ=80
κ=100
Segment Prompts
clear blue sky
city skyline with lot of skyscrapers
dark blue sea
Global Prompt
city skyline with lot of skyscrapers besides dark blue sea under clear blue sky

Figure 8. A visual comparison of the generations using *segment-specific prompts* and *global prompts* for the Harmonization stage. *Harmony:* Our results show that both achieve comparable harmony with global prompts having a slight edge. *Detailing:* For detailing within a segment, the segment-specific prompts provide a slight edge. Since both these methods apply only to the harmonization stage, for lower scaffolding values (e.g. $\kappa = 0, 20$), the outputs vary noticeably, while at the higher values, since the number of steps for diffusion is reduced, the outputs are very close to each other.

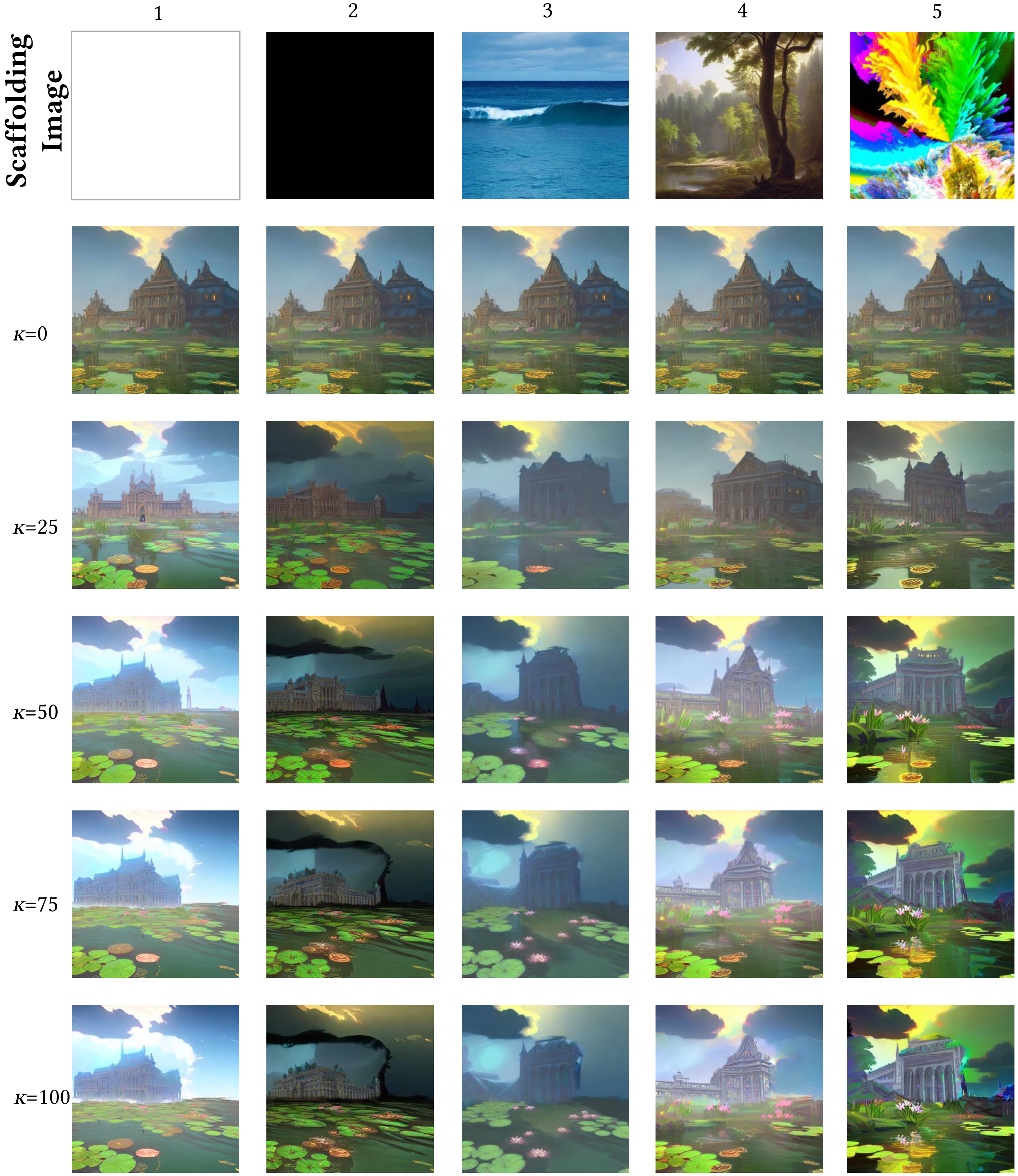


Figure 9. Effect of Scaffolding Image and Scaffolding factor (κ) on Segment Layout Fidelity and Blending & Harmony. Shown are generations for different scaffolding images and for 5 different scaffolding factors using the same seed value, the same number of ddim steps (50), and the same $\eta = 0$ value for the running example 3.1. In general, the conformance to the spatial layout increases with the increase in scaffolding value, while there is a decrease in the segment blending and harmonization. The effects vary for different scaffolding images.

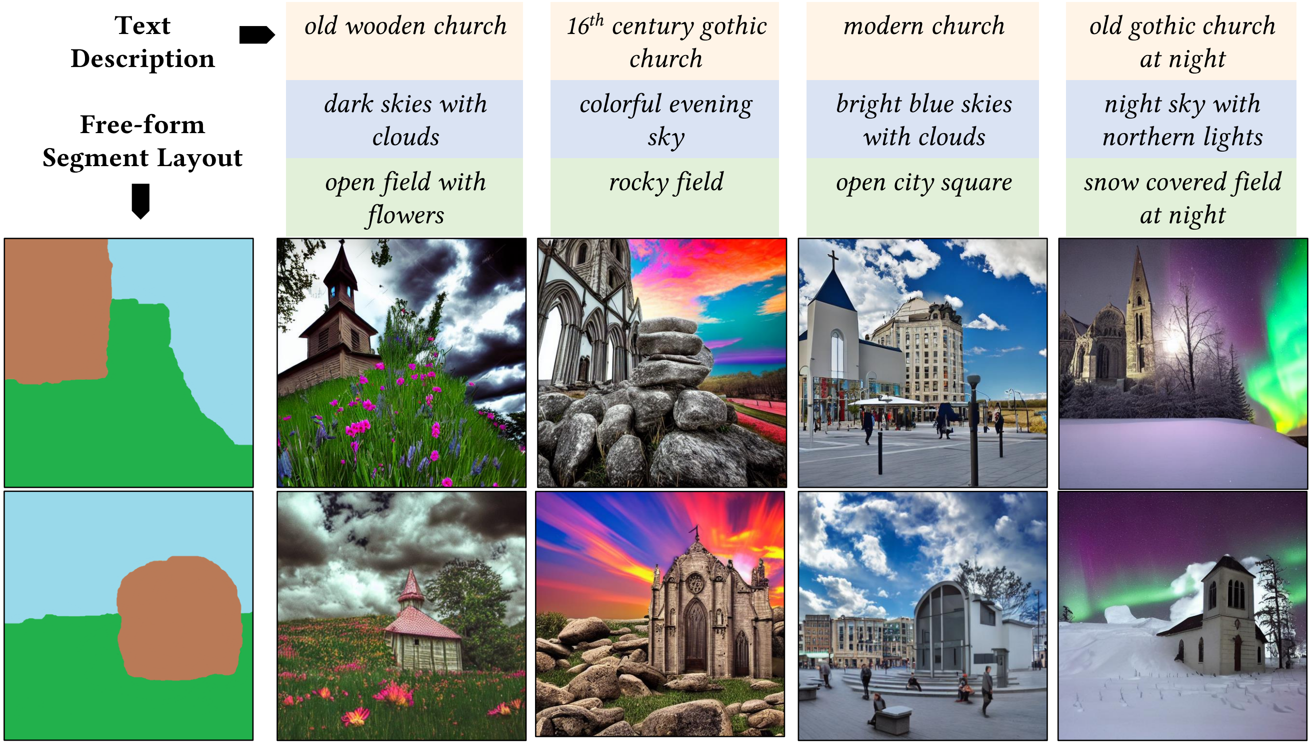


Figure 10. By controlling layout, and/or text inputs independently an artist can produce diverse pictures through Composite diffusion methods. Note how the segment layout is used as a guide for *sub-scenes* within an image and not as an outline of shapes for the objects as happens in many object segment models.

very well on closeup faces of human beings but the faces may get a bit distorted when we generate a full-length picture of a person or a group of people.

5.5. Societal impact

Recent rapid advancements in generative models have been so stunning that they have left many people in society (and in particular, the artists) both worried and excited at the same time. On one hand, these tools, especially when they are getting increasingly democratized and accessible, give artists an enabling tool to create powerful work in lesser time. On the other hand, traditional artists are concerned about losing the business critical for their livelihood to amateurs [35]. Also, since these models pick off artistic styles easily from a few examples, the affected artists, who take years to build their portfolio and style, might feel shortchanged. Also, there is a concern that AI art maybe be treated at the same level and hence compete with traditional art.

We feel that generative AI technology is as disruptive as photography was to realistic paintings. Our work, in particular, is based on Generative Models that can add to the consequences. However, since our motivation is to help artists improve their workflow and create images that self-express them, this modality of art may also have a very positive impact on their art and art processes. With confidence tempered with caution, we believe that it should be a welcome addition to an artist's toolkit.

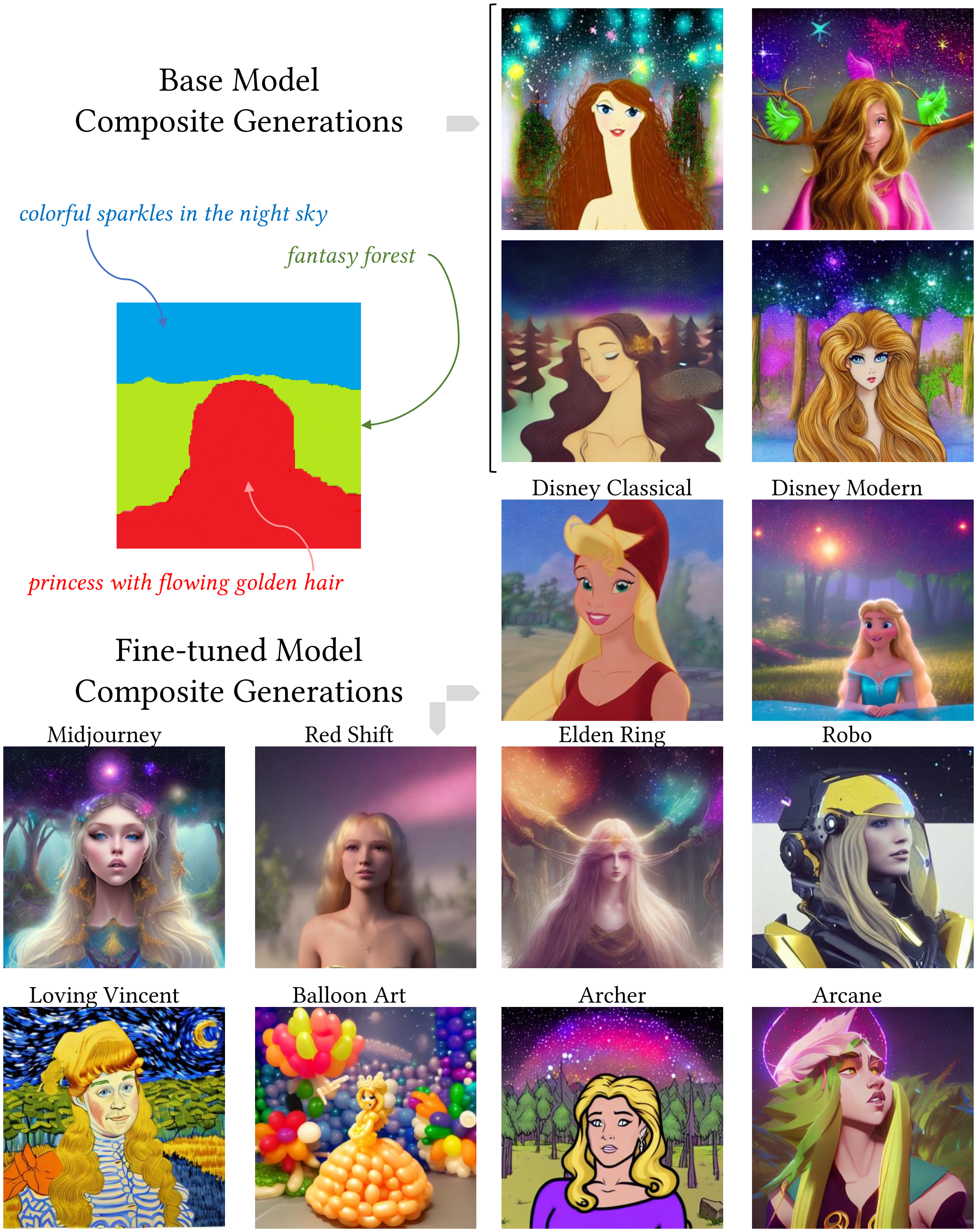


Figure 11. Composite generations using fine-tuned models. Using the same layout and same captions, but different specially trained fine-tuned models, the generative artwork can be customized to a particular style or artform. Note that our Composite Diffusion methods are plug-and-play compatible with these different fine-tuned models.

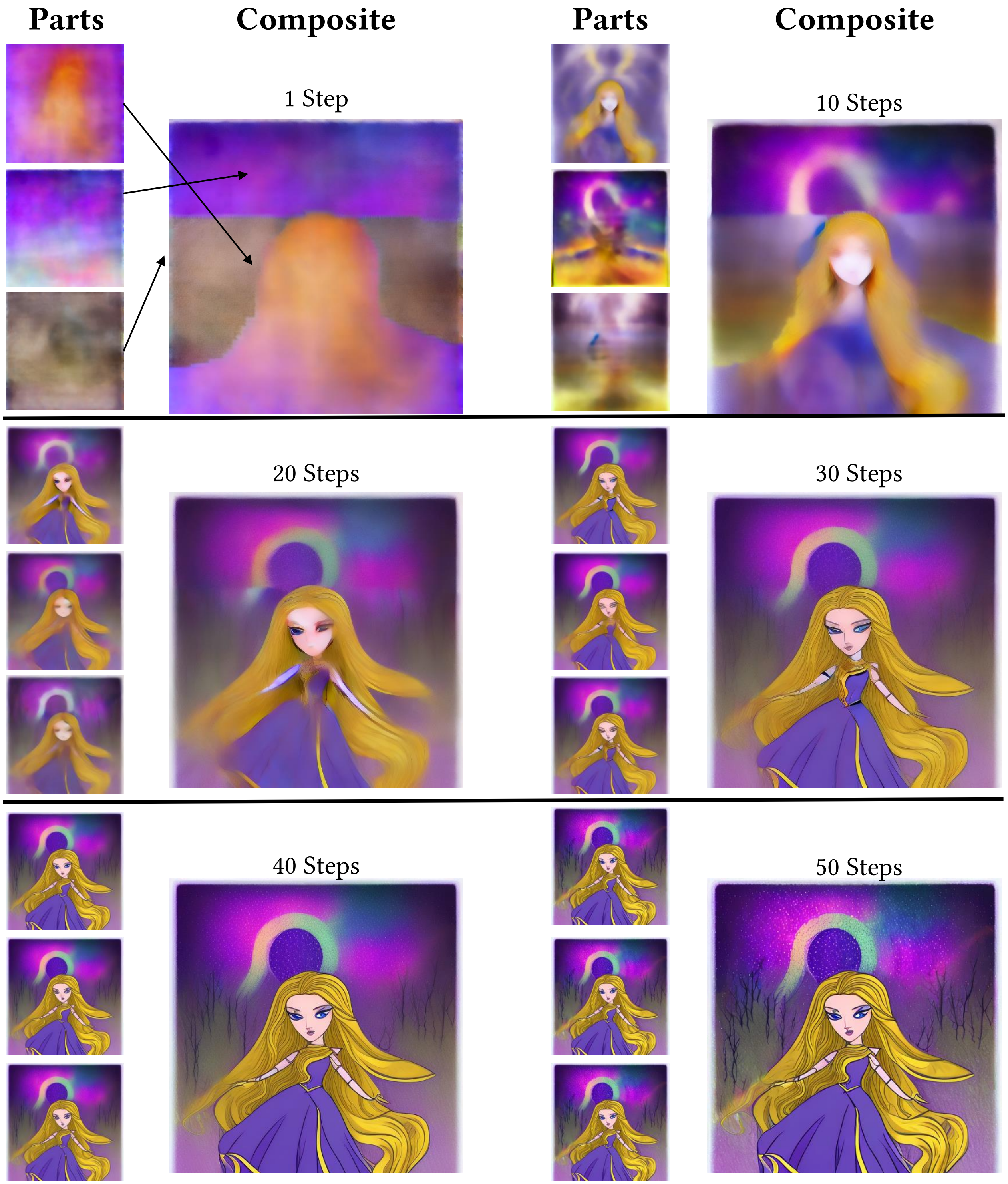


Figure 12. Composite Diffusion generation using the inputs specified in Fig. 11, a scaffolding factor of $\kappa = 30$, and 50 DDIM diffusion steps. The figure shows segment latents and composites after the timesteps 1, 10, 20, 30, 40, and 50. Note that for the first 15 steps (scaffolding stage), the segment latents develop *independently*, while for the remaining 35 steps (harmonization stage), the latents develop *in-the-context* of all other segments.

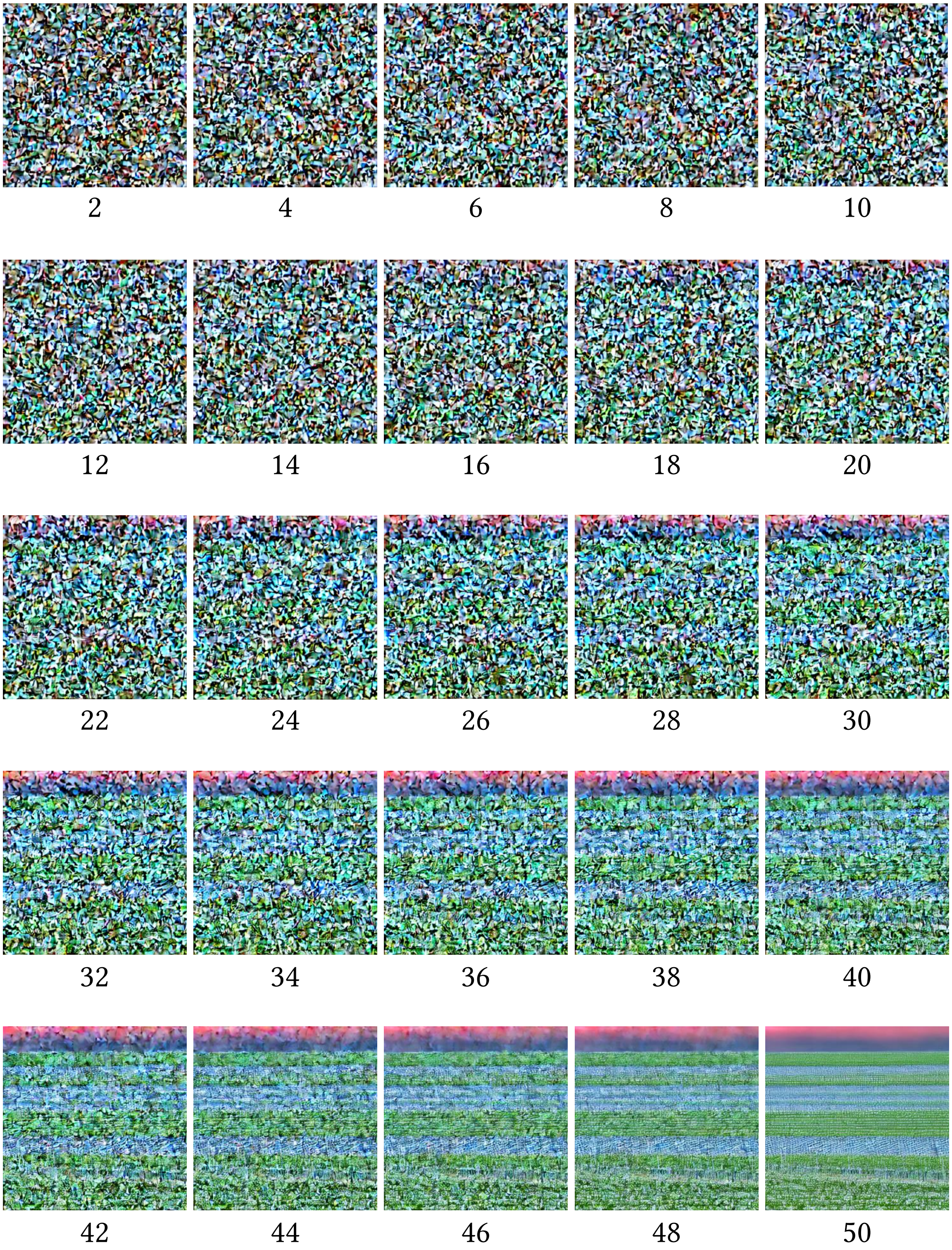


Figure 13. Composite generation using **Serial Inpainting**. The figure shows the development stages for the **Segment 1**. The inputs to the model are as shown in the running example of Fig. 1.

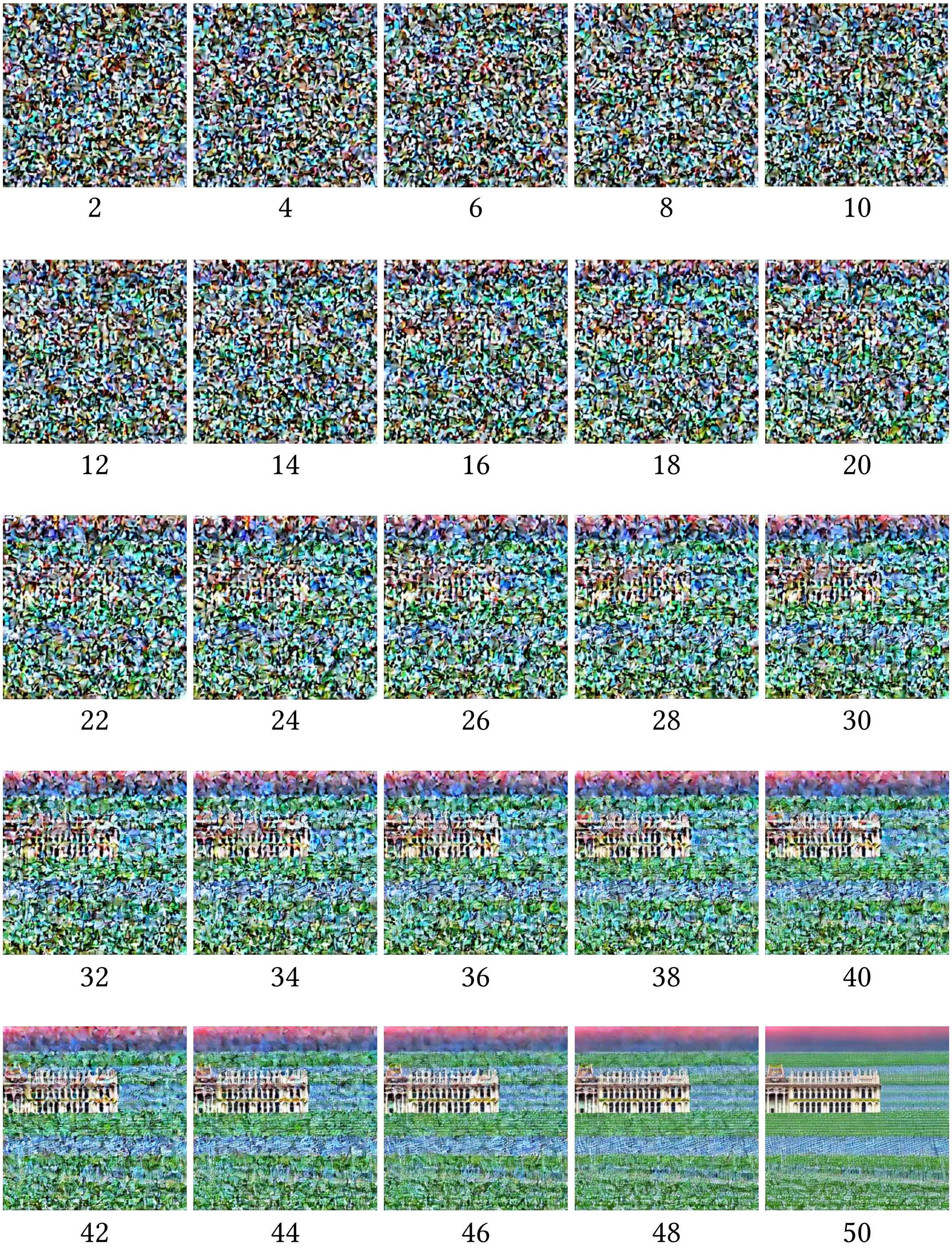


Figure 14. Composite generation using **Serial Inpainting**. The figure shows the development stages for the **Segment 2**. The inputs to the model are as shown in the running example of Fig. 1.

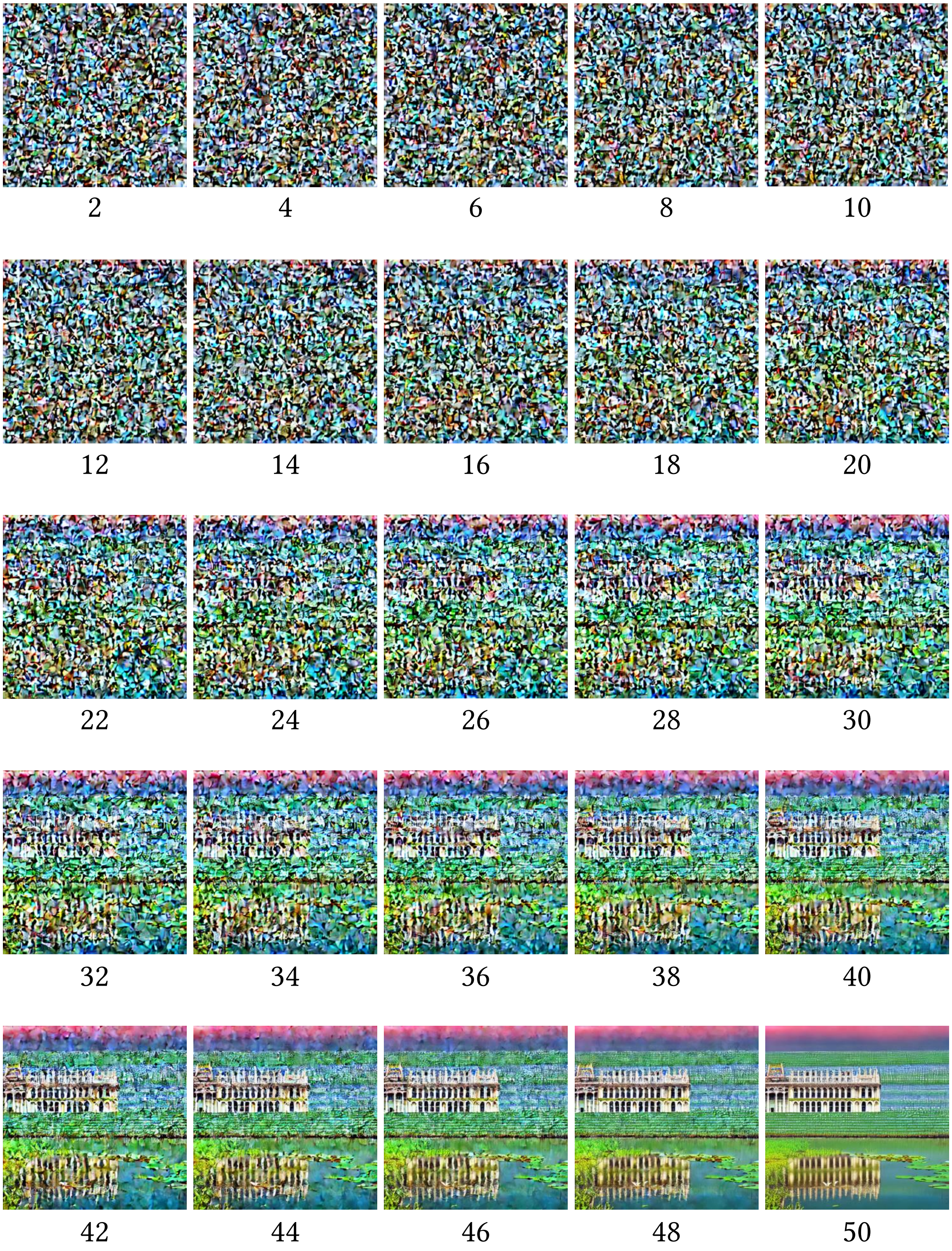


Figure 15. Composite generation using **Serial Inpainting**. The figure shows the development stages for the **Segment 3**. The inputs to the model are as shown in the running example of Fig. 1.

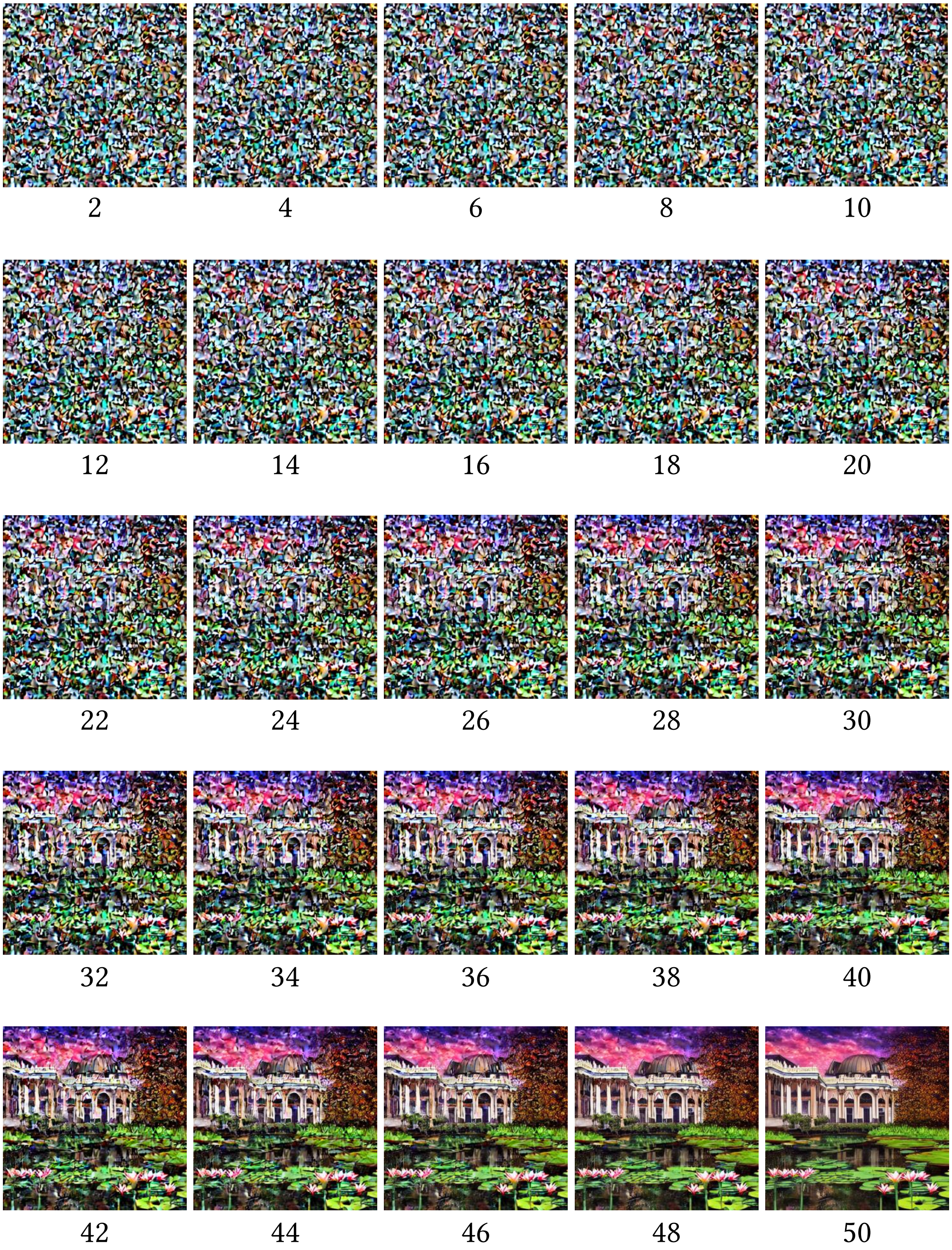


Figure 16. Composite generation using **Composite Diffusion**. The figure shows the development stages of the composite image. The inputs to the model are as shown in the running example of Fig. 1.

6. Detailed Related work

In this section, we discuss the approaches that are related to our work from multiple perspectives.

6.1. Text-to-Image generative models

The field of text-to-image generation has recently seen rapid advancements, driven primarily by the evolution of powerful neural network architectures. Approaches like DALL · E [39] and VQ-GAN [16] proposed a two-stage method for image generation. These methods employ a discrete variational auto-encoder (VAE) to acquire comprehensive semantic representations, followed by a transformer architecture to autoregressively model text and image tokens. Subsequently, diffusion-based approaches, such as Guided Diffusion [31] [13], have showcased superior image sample quality compared to previous GAN-based techniques. Dalle-2 [38] and Imagen [43] perform the diffusion process in the pixel-image space while Latent Diffusion Models such as Stable Diffusion [40] perform the diffusion process in a more computationally suitable latent space. However, in all these cases, relying on single descriptions to depict complex scenes restricts the level of control users possess over the generation process.

6.2. Spatial control models

Some past works on image generation have employed segments for spatial control but were limited to domain-specific segments. For example, GauGAN [33] introduced spatially-adaptive normalization to incorporate semantic segments to generate high-resolution images. PoE-GAN [23] utilized the product of experts method to integrate semantic segments and a global text prompt to enhance the controllability of image generation. However, both approaches rely on GAN architectures and are constrained to specific domains with a fixed segment vocabulary. Make-A-Scene [17] utilized an optional set of dense segmentation maps, along with a global text prompt, to aid in the spatial controllability of generation. VQ-GAN [16] can be trained to use semantic segments as inputs for image generation. No-Token-Left-Behind [32] employed explainability-based methods to implement spatial conditioning in VQ-GAN; they propose a method that conditions a text-to-image model on spatial locations using an optimization approach. The approaches discussed above are also limited by training only on a fixed set of dense segments.

6.3. Inpainting

The work that comes closest to our approach in diffusion models is in-painting. Almost all the popular models [38], [43], [40] support some form of inpainting. The goal of inpainting is to modify a portion in an image specified by a segment-mask (and optional accompanying textual description) while retaining the information outside the segment. Some of the approaches for inpainting in the recent past include repaint [27], blended-diffusion [5], and latent-blended diffusion [3]. RunwayML [40] devises a specialized model for in-painting in Stable Diffusion, by modifying the architecture of the UNet model to include special masked inputs. As we show in later this paper, one can conceive of an approach for Composite Diffusion using inpainting, where we can perform inpainting for each segment in a serial manner (refer to Appendix 4). However, as we explain in this paper, a simple extension of localized in-painting methods for multi-segment composites presents some drawbacks.

6.4. Other diffusion-based composition methods

Some works look at the composition or editing of images through a different lens. These include prompt-to-prompt editing [19, 29], composing scenes through composable prompts [25], and methods for personalization of subjects in a generative model [42]. Composable Diffusion [26] takes a structured approach to generate images where separate diffusion models generate distinct components of an image. As a result, they can generate more complex imagery than seen during the training. Composed GLIDE [25] is a composable diffusion implementation that builds upon the GLIDE model [30] and utilizes compositional operators to combine textual operations. Dreambooth [42] allows the personalization of subjects in a text-to-image diffusion model through fine-tuning. The learned subjects can be put in totally new contexts such as scenes, poses, and lighting conditions. Prompt-to-prompt editing techniques [12, 19, 29] exploit the information in cross-attention layers of a diffusion model by pinpointing areas that spatially correspond to particular words in a prompt. These areas can then be modified according to the change of the words in the prompt. Our method is complementary to these advances. We concentrate specifically on composing the spatial segments specified via a spatial layout. So, in principle, our methods can be supplemented with these capabilities (and vice versa).

6.5. Spatial layout and natural text-based models

In this section, we discuss three related concurrent works: SpaText [4], eDiffi [6], and Multi-diffusion [7]. All these works provide some method of creating images from spatially free-form layouts with natural text descriptions.

SpaText [4] achieves spatial control by training the model to be space-sensitive by additional CLIP-based spatial-textual representation. The approach requires the creation of a training dataset and extensive model training, both of which are costly. Their layout schemes differ slightly from ours as they are guided towards creating outlines of the objects, whereas we focus on specifying the sub-scene.

eDiffi [6] proposes a method called paint-with-words which exploits the cross-attention mechanism of U-Net in the diffusion model to specify the spatial positioning of objects. Specifically, it associates certain phrases in the global text prompt with particular regions by manipulating the cross-attention matrix. Similar to our work, they do not require pre-training for a segment-based generation. However, they must create an explicit control for the objects in the text description for spatial control. We use the inherent capability of U-net's cross-attention layers to guide the relevant image into the segments through step-inpainting and other techniques.

Multi-diffusion [7] proposes a mechanism for controlling the image generation in a region by providing the abstraction of an optimization loss between an ideal output by a single diffusion generator and multiple diffusion processes that generate different parts of an image. It also provides an application of this abstraction to segment layout and natural-text-based image generation. This approach has some similarities to ours in that they also build their segment generation by step-wise inpainting. They also use bootstrapping to anchor the image and then use the later stages for blending. However, our approach is more generic, has a wider scope, and is more detailed. For example, we don't restrict the step composition to a particular method. Our scaffolding stage has a much wider significance as our principal goal is to create segments independent of each other, and the goal of the harmonization stage is to create segments in the context of each other. We provide alternative means of handling both the scaffolding and harmonization stages.

Further, in comparison to all the above approaches, we achieve *additional control over the orientation and placement of objects within a segment* through reference images and control conditions specific to the segment.

Figures 17, 18, and 19 further provide a visual comparison of our Composite Diffusion methods with the baselines discussed in this paper.

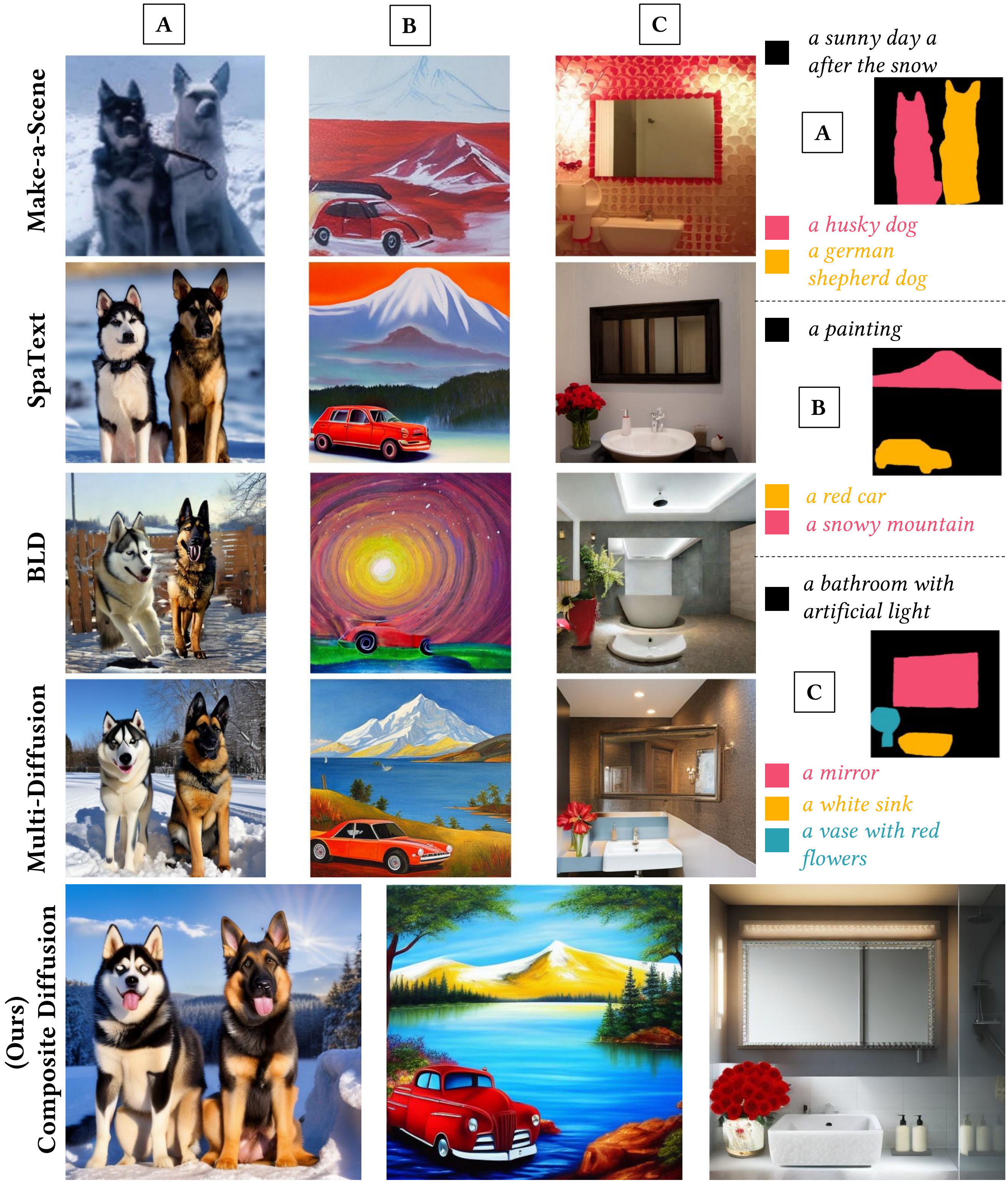


Figure 17. The figure provides a visual comparison of the outputs of Composite Diffusion with other related approaches - using the same segment layouts and text prompts. Considered approaches are: Make-a-Scene [17], SpaText [4], Blended Diffusion [3], and Multi-diffusion [7]. Note that these input specifications are from the related-work literature [4, 7]. Given a choice, our approach to creating segment layout and text prompts would vary slightly - we would partition the image space into distinct *sub-scenes* that fully partition the image space, and we will not have background masks or prompts.

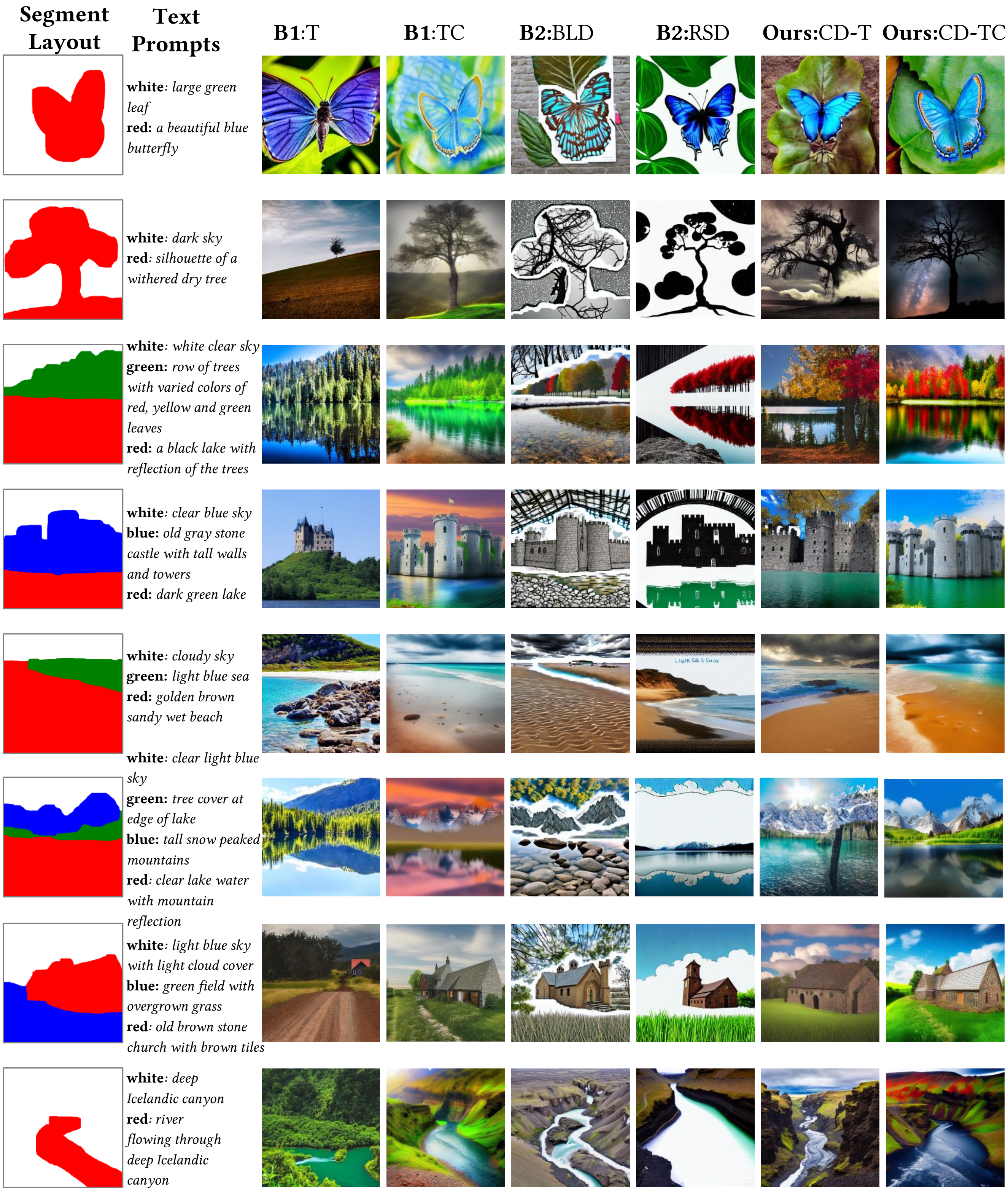

Segment Layout
Text Prompts
B1:T
B1:TC
B2:BLD
B2:RSD
Ours:CD-T
Ours:CD-TC
white: large green leaf
red: a beautiful blue butterfly
white: dark sky
red: silhouette of a withered dry tree
white: white clear sky
green: row of trees with varied colors of red, yellow and green leaves
red: a black lake with reflection of the trees
white: clear blue sky
blue: old gray stone castle with tall walls and towers
red: dark green lake
white: cloudy sky
green: light blue sea
red: golden brown sandy wet beach
white: clear light blue sky
green: tree cover at edge of lake
blue: tall snow peaked mountains
red: clear lake water with mountain reflection
white: light blue sky with light cloud cover
blue: green field with overgrown grass
red: old brown stone church with brown tiles
white: deep Icelandic canyon
red: river flowing through deep Icelandic canyon

Figure 18. Example visual comparisons with baselines B1(base models) and B2 (serial inpainting). The considered base model approaches are: B1:T, text-to-image Stable Diffusion model [40], and B1:TC, Stable Diffusion implementation of Controlnets [49]. The considered approaches for serial inpainting B2:BLD and B2:RSD are respectively based on inpainting approaches of blended diffusion [3], and specialized Runway ML inpainting method for Stable Diffusion [40].

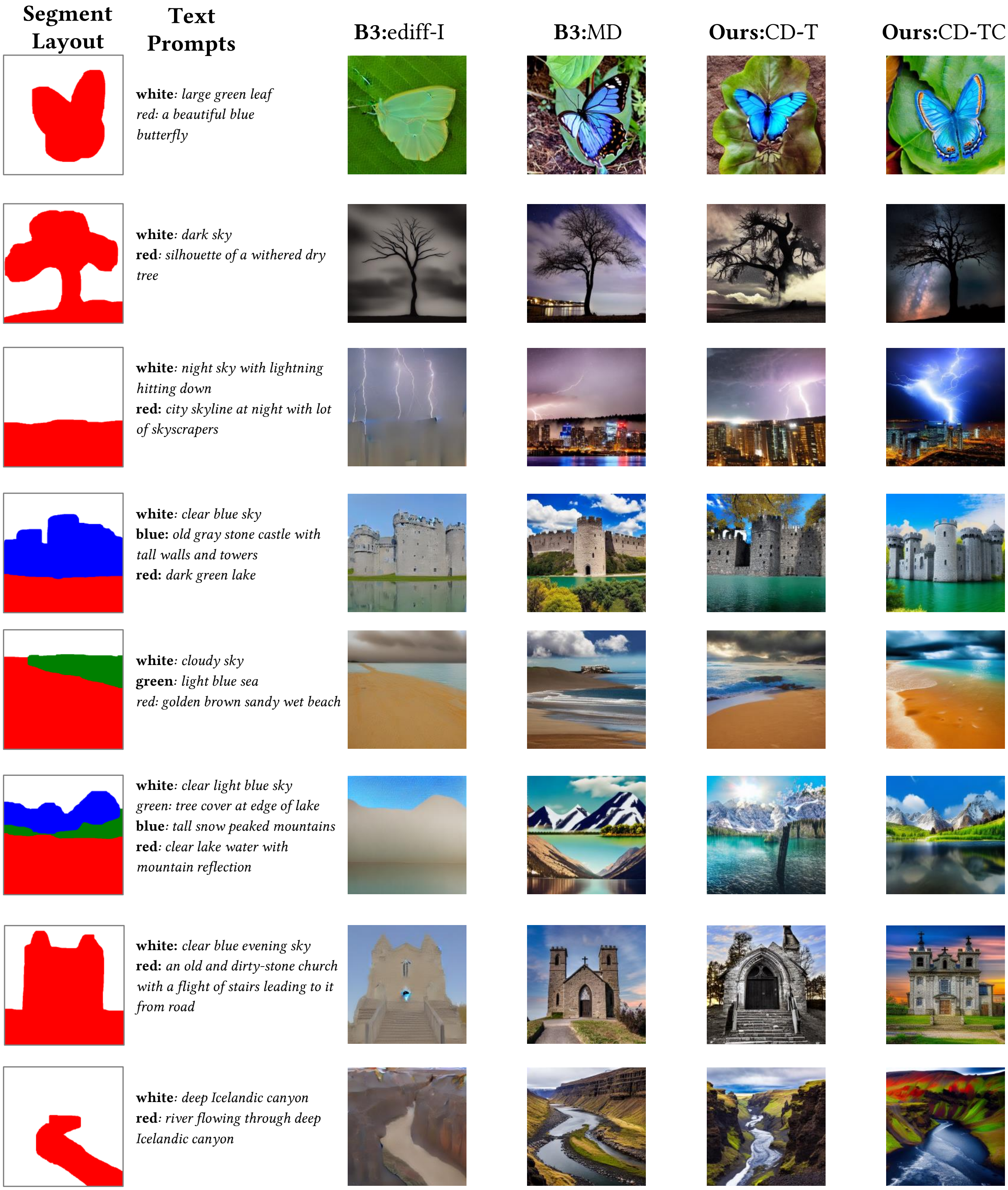

Segment Layout
Text Prompts
B3:ediff-I
B3:MD
Ours:CD-T
Ours:CD-TC
white: large green leaf
red: a beautiful blue butterfly
white: dark sky
red: silhouette of a withered dry tree
white: night sky with lightning hitting down
red: city skyline at night with lot of skyscrapers
white: clear blue sky
blue: old gray stone castle with tall walls and towers
red: dark green lake
white: cloudy sky
green: light blue sea
red: golden brown sandy wet beach
white: clear light blue sky
green: tree cover at edge of lake
blue: tall snow peaked mountains
red: clear lake water with mountain reflection
white: clear blue evening sky
red: an old and dirty-stone church with a flight of stairs leading to it from road
white: deep Icelandic canyon
red: river flowing through deep Icelandic canyon

Figure 19. Example visual comparison of Composite Diffusion with B3 related work baseline. Approaches considered are publicly available implementations of B3:ediff-I [6] paint-by-word method and B3:Multi-diffusion [7].

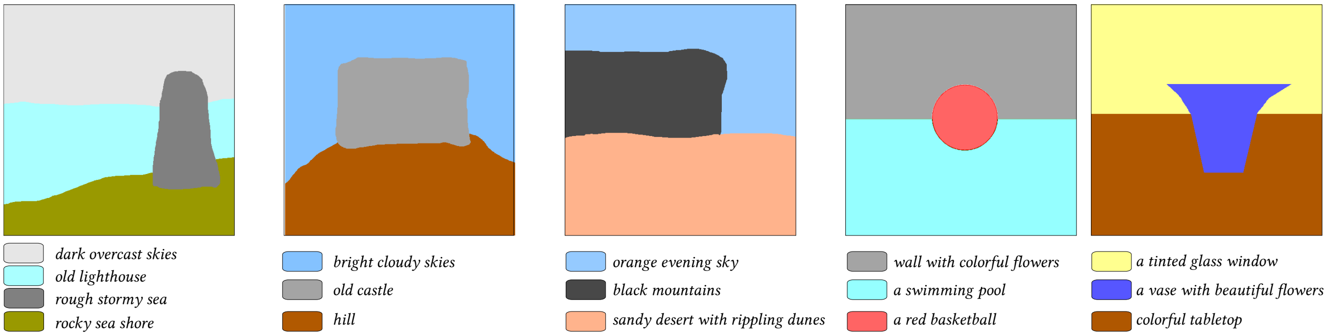


Figure 20. Segment layouts and segment text prompts as inputs for Survey sample generations

7. Human evaluation and survey details

During the course of the project, we conducted a set of three different surveys. A preliminary and a revised survey were conducted on the general population, and a separate survey was conducted on artists and designers. In these surveys, we evaluated the generated outputs of text-to-image generation, serial inpainting generation methods, and our composite generation methods. The text-to-image generations serve as the first baseline (**B1**) and serial inpainting generations serve as the second baseline (**B2**) for comparison.

7.1. Survey design

The survey design involved the following parts:

7.1.1 Method for choosing survey samples

We designed five different input sets for the survey. The free-form segment layouts and corresponding text descriptions were chosen to bring in a variety of scenarios for the input cases. Refer to Fig. 20 for the inputs. Since we also wanted to compare our generations with the base model generations, and text-to-image model only allows a single text prompt input, we manually crafted the prompts for the first base case. This was done by: (i) creating a text description that best tries to capture the essence of different segment prompts, (ii) concatenating the different segment descriptions into a single text description. The best of these two generations were taken as the representative pictures for base models. For selecting the samples from the different algorithms we followed the following protocol. Since the underlying models are of different architecture (for example, the serial inpainting method uses a specialized inpainting model and requires a background image), we generated 3 images using random seeds for each algorithm and for each set of inputs. We then chose the best representatives (1 out of 3) from each algorithm for the survey samples.

7.1.2 Survey questions:

In each of the surveys, the survey takers were presented with a Google form on the web containing anonymized and randomly sorted images generated from these three algorithms with corresponding inputs. The respondents were asked to rate each of these images on five quality parameters. We explained each quality parameter and asked a corresponding quality question as listed below:

1. *Text Fidelity:* How closely does the image match the text prompts?
2. *Mask Fidelity:* How closely does the image match the mask shapes?
3. *Blending & Harmony:* How well do the segments blend together and how harmonious is the overall image?
4. *Technical Quality:* How would you rate the overall technical quality of the image?
5. *Aesthetic Quality:* How would you rate the overall aesthetic quality of the image?

The respondents were asked to rate a generated image for a given quality parameter on a scale of 1 to 5 (semantic differential scales). We also provided a rough rating guideline for these parameters. Refer to Fig.21 for a snapshot of the web survey.

7.2. Survey execution:

The details of the execution of the three surveys are as follows:

7.2.1 Phase 1: Preliminary survey:

We conducted this survey on a diverse set of 14 respondents who were spread across age (20-80), gender, and profession. Our experience with the first survey gave us vital feedback on how to design the survey

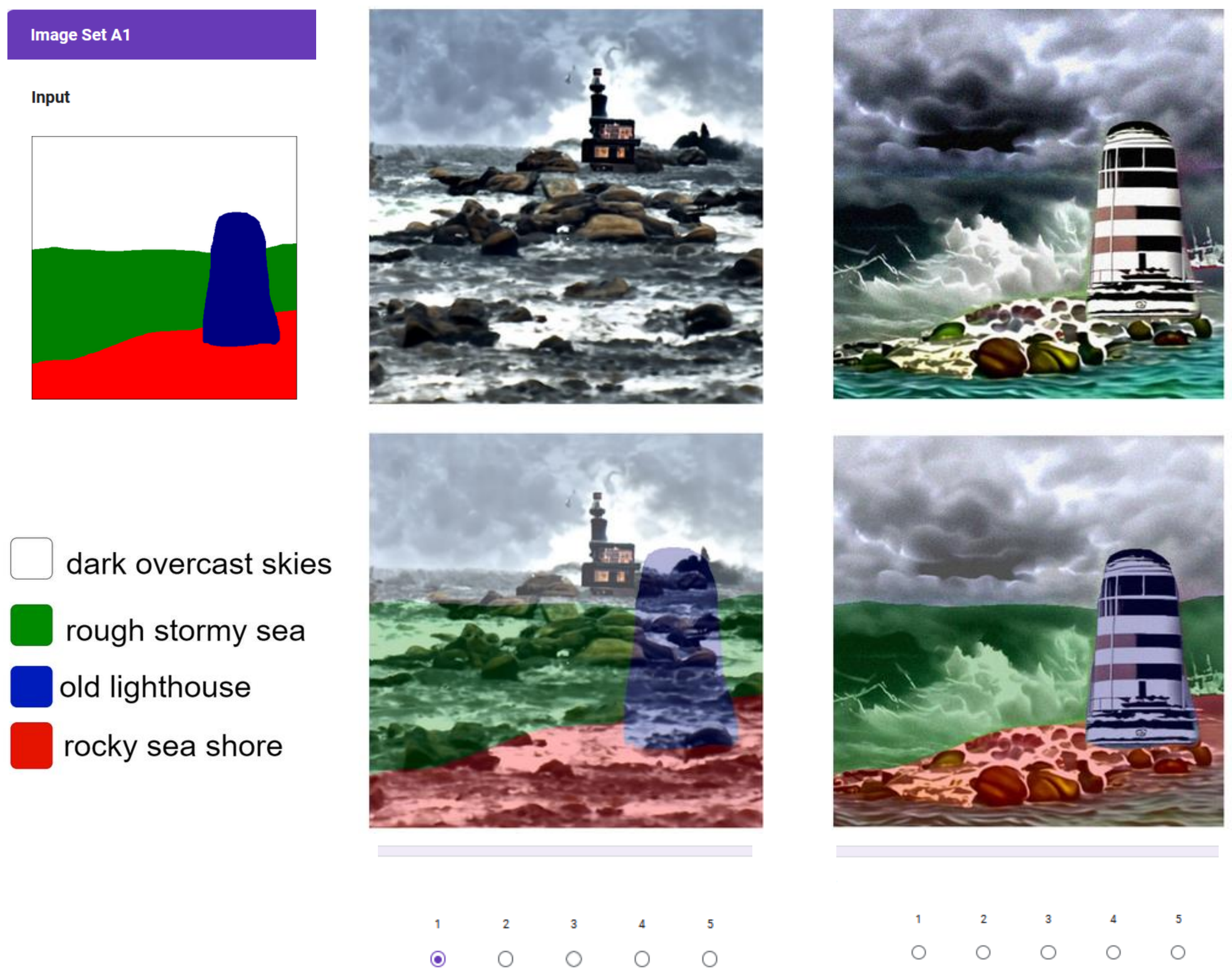


Image Set A1

Input

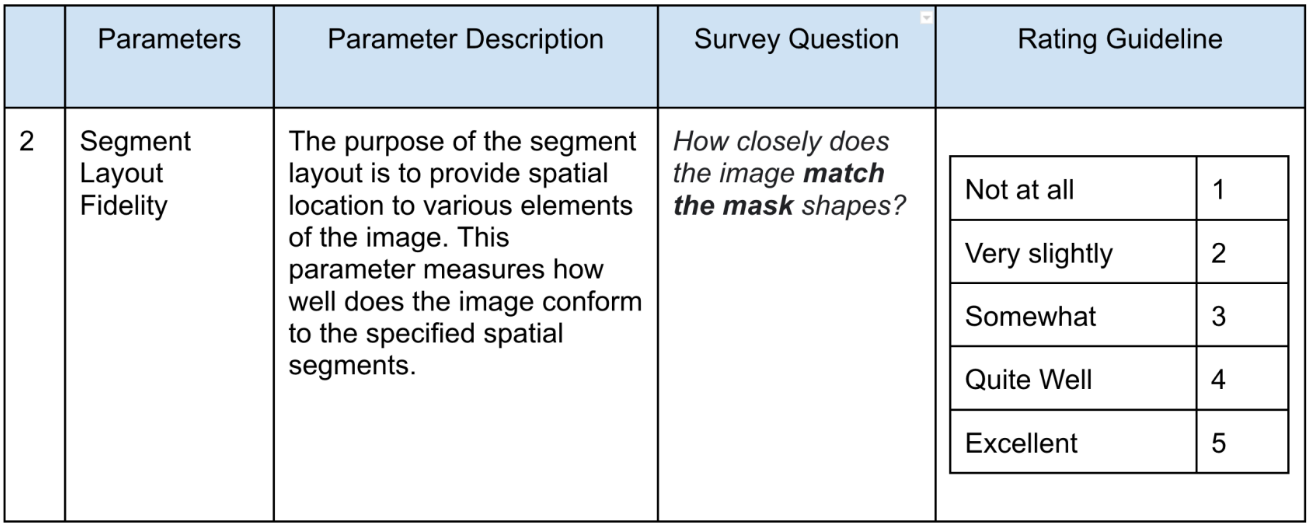
⬜ dark overcast skies

🟩 rough stormy sea

🟦 old lighthouse

🟥 rocky sea shore

	1	2	3	4	5
	●	○	○	○	○
	○	○	○	○	○



	Parameters	Parameter Description	Survey Question	Rating Guideline
2	Segment Layout Fidelity	The purpose of the segment layout is to provide spatial location to various elements of the image. This parameter measures how well does the image conform to the specified spatial segments.	*How closely does the image **match the mask** shapes?*	Not at all 1 Very slightly 2 Somewhat 3 Quite Well 4 Excellent 5

Figure 21. Snippets from the interface used for collecting responses in user evaluation

more effectively. For example, many surveyors said that they found it tough to take the survey as it was lengthy. There were a total of 75 rating questions that needed to be answered. So there was some fatigue due to the cognitive load. The first survey was organized in the following manner: Each set of inputs was a separate page and contained all five quality questions. On each page, the respondents were presented with 3 pics from 3 different algorithms(anonymized and randomly sorted) and were asked to rate each of the pictures on five quality parameters. We also received feedback that all the guidance information was on the front page, and they had to revisit it several times to understand the rating guidelines and the meaning of each quality parameter. Further, some users told us that 'aesthetics' influenced their rating of the other qualities; They tended to rate an image with higher aesthetics higher for other qualities as well.

7.2.2 Phase 2: Revised survey

We built upon this feedback, and without changing the content, restructured the survey to make it more modular for our final assessment. We also found the guidelines in [10] relevant and followed them to fine-tune the survey organization. The two major changes were: (1) Each quality parameter was made into a separate survey. This was done to help the surveyors focus on one quality parameter at a time. (2) We provided guidelines for the score rating on each of the survey pages as a ready reference.

The survey was further divided into two sets of Surveyors representing different sets of professional skills.

- **Survey population: Artists and Designers (AD)**: We conducted this survey during the final phase of our project. We used the same set of images as used in the preliminary survey to collect responses from artists and designers. We took the help of Amazon M-Turk for collecting responses for this survey. There was no restriction on whether a person took all 5 surveys or only a subset of them. There were a total of 20 respondents for each of the five surveys (where one survey was comprised of a distinct quality parameter).
- **Survey population: General (GP)**: We conducted this survey simultaneously with the above survey. The participants in this survey were chosen from a larger general population that also included professionals such as engineers and software developer. In this case, 22 respondents completed all the five survey sets, while 48 completed at least one set.

Method	**B1**	**B2**	**Ours**
Content Fidelity ↑	3.81±1.0	3.22±1.07	3.92±0.97
Spatial Layout Fidelity ↑	3.21±1.08	3.14±1.15	3.62±0.97
Blending & Harmony ↑	3.87±0.96	4.02±0.99	3.86±1.03
Technical Quality ↑	3.85±1.02	3.75±1.15	3.6±0.98
Aesthetic Quality ↑	3.55±0.92	3.55±1.0	3.52±0.99

Table 2. Results of the survey conducted on Artists and Designers

Method	**B1**	**B2**	**Ours**
Content Fidelity ↑	2.8±1.28	2.38±1.13	3.12±1.45
Spatial Layout Fidelity ↑	2.19±1.11	2.99±1.44	3.82±1.08
Blending & Harmony ↑	3.47±1.07	2.94±1.22	3.62±1.14
Technical Quality ↑	3.33±1.11	2.78±1.16	3.39±1.14
Aesthetic Quality ↑	3.16±1.19	2.66±1.26	3.36±1.28

Table 3. Results of the survey conducted on General Population.

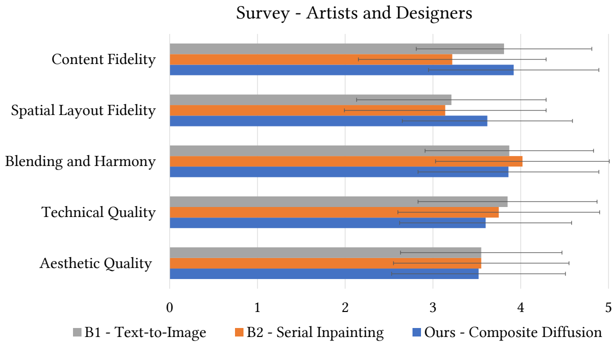


Figure 22. Human evaluation results from the set - Artists/Designers(AD)

7.3. Additional rurvey results

Table 2 presents the results of the survey for the artists and designers population, and Fig. 22 presents a graphical representation of the same for easy comparison. Since the set of images and the survey questions were the same across the two phases of the survey, we consolidated the results of general population responses. Table 3 presents the consolidated results of the survey of the general population, and Fig. 6 in the main paper gives a graphical representation of the same.

Kappa	Content Fidelity ↑	Spatial Layout Fidelity ↑	Technical Quality ↑	Human Preference ↓	Aesthetic Score↑	Blending & Harmony ↓
0	0.2634	0.278	1.2612	3	6.1809	5321
20	0.2629	0.278	1.2079	3	6.1487	6137
40	0.2596	0.2726	1.6987	3	6.296	8078
60	0.2627	0.2757	1.4186	4	6.2565	6827
80	0.2594	0.2744	1.3123	4	5.9693	7235
100	0.2579	0.2773	1.7702	3	6.0798	7699

Table 4. Automated Method evaluation across different scaffolding factor κ values. We observe that the general trend is that Blending & Harmony (lower is better) progressively gets slightly worse as we move from lower to higher κ, while the other factors remain quite similar across different κ values.

8. Automated evaluation methods

We find that the present methods of automated quality comparisons such as FID and IS aren't well suited for the given quality criteria. In the section below we discuss a few of the methods that are widely used in measuring the capabilities of generative models, point out their drawbacks, and then detail our methods for automated evaluation.

8.1. Current approaches for automated evaluation

Inception score (IS), Fréchet inception distance (FID), precision, and recall are some of the commonly used metrics for assessing the quality of synthetically generated images [9,20,44,45]. IS score jointly measures the diversity and quality of generated images. FID measures the similarity between the distribution of real images and generated images. Metrics like precision and recall [44] separately capture the quality and diversity aspects of the generator. Precision is an indicator of how much the generated images are similar to the real ones, and recall measures how good the generator is in synthesizing all the instances of the training data set [9].

These approaches have some drawbacks to our requirement of assessing the quality of Composite Diffusion generations: (i) These approaches require a large set of reference images to produce a statistically significant score. The distribution of the training set images is not relevant to us. We need datasets that have - an input set of sub-scene layouts along with textual descriptions of those sub-scenes, and a corresponding set of reference images., (ii) Even if we had the facility of a relevant large dataset, these methods assume that the reference images provide the highest benchmark for quality and diversity. This might not be always true as the generated images can exceed the quality of reference images and have a variety that is different from the reference set., and (iii) These methods don't measure the quality with the granularity as described in the quality criteria that we use in this paper.

8.2. Our approach for automated evaluation

We devise the following automated methods to evaluate the generated images based on our quality criteria.

Content Fidelity ↑: The objective here is to obtain a measure of how similar the image is to each of the artist's intended content, and in this case, we use the textual descriptions as content. We compute the cosine similarity between the CLIP embeddings of the image and the CLIP embeddings of each segment's description. We then take the mean of these similarity scores. Here *a greater score indicates greater content fidelity.*

Spatial-layout Fidelity ↑: The objective here is to measure how accurately we generate a segment's content. We use masking to isolate a segment from the image. We find the CLIP similarity score between the masked image and that segment's description. We do this for all the segments and then take the mean of these scores. Here *a greater score indicates greater spatial-layout fidelity.*

Technical Quality ↓: The goal here is to measure if there are any degradations or the presence of unwanted artifacts in the generated images. It is difficult to define all types of degradations in an image. We consider the presence of noise as a vital form of degradation. We estimate the Gaussian noise level in the image by using the method described in [11]. Here *a lower score indicates greater technical quality.*

Aesthetics ↑: We use the aesthetic quality estimator from [24] to get an estimate of the aesthetic quality of the image. This method uses a linear layer on top of the CLIP embedding model and is trained on 4000 samples to estimate if an image is looking good or not. Here *a greater score indicates greater perceived aesthetics.*

Blending & Harmony ↓: We detect the presence of edges around the segment boundaries as a measure of better blending. Hence *a lower value in this case indicates better blending.*

Human Preference ↓: To additionally estimate the human preference we rank the images generated by the different algorithms using ImageReward [48]. This method uses a data-driven approach to score human preferences for a set of images. Here *a greater score indicates lower preference.*

8.3. Limitations of automated evaluation methods

As stated in the main paper, these measures are the initial attempts and may give only a ballpark estimation of the qualities under consideration. Content Fidelity and Spatial-layout metrics are only as good as the capability underlying the image-text model - OpenAI's CLIP model [37]. Technical quality should give an overall measure of technical aberrations like color degradation, unwanted line artifacts, etc. However, we limit ourselves to only measuring the overall noise levels. Aesthetics is a highly subjective aspect of image quality and the CLIP aesthetic model [24], though effective, has been trained on a relatively small-sized dataset. Blending & Harmony in our case is limited to measuring the presence of edges around the boundaries of a segment. Measuring harmony in images is a challenging problem as one needs to also consider the positioning, scale, and coloring of the elements and segments in the context of the overall image. Human preference scoring utilizes ImageReward [48], which models the ranking that humans would assign to a group of images. Although this method performs better than CLIP and BLIP in this aspect, it lacks the explainability of why one image is ranked higher over the other.

Finding better, more precise, and holistic machine-assisted methods for measuring the qualities presented in this paper is an opportunity for future research.

8.4. Benchmark dataset

A notable challenge in the automated evaluation of the composite diffusion method is the lack of benchmark datasets. Currently, there do not exist any datasets that consist of segment (or sub-scene) layouts with rich textual descriptions for each segment. Creation of such a dataset is non-trivial using automated methods and requires expensive annotation [4].

We handcraft a dataset containing 100 images where we segment each representative image into sub-scenes and manually annotate each sub-scene with a relevant textual description. This enables us to build a benchmark dataset for composite image generation with sufficiently high-quality data. We use this dataset to generate images using our baseline and Composite Diffusion methods. We use the automated methods described above to get the automated evaluation results (Table 4). We initially gathered 100 diverse images from Google Images, ensuring each was available under a commercial license. Using these images as a reference, we construct free-form spatial masks on them and craft textual prompts for each of these masks.

Additionally, we created a control condition dataset for the evaluation of Controlnet base models and the evaluation of the Composite Diffusion approach using control conditions. This was obtained by processing all the reference images through a Lineart preprocessor. To obtain the segment-specific control conditions, we then segmented the reference control condition images with the help of respective segment masks.

9. Results and discussion

In this section, we summarize the results from the different types of evaluations and provide our analysis for each quality criterion.

9.1. Content fidelity

Ours:CD-TC and Ours:CDT get the highest content fidelity scores followed by B3:MD, B3:ediff-I and B2:BLD and B2:RSD. B1:TC and B1:T get the least scores.

Our take: The performance of Ours, B3, and B2 can be attributed to the rich textual descriptions used for describing each image segment, resulting in an overall increase in semantic information and control in the generation process. Moreover, the explicit scaffolding stage in Our method leads to more conformity to the textual descriptions compared to B3. One can argue that similar rich textual descriptions are also available for the serial inpainting method (B2). However, B2 might get several limitations: (i) There is a dependency on the initial background image that massively influences the inpainting process, (ii) There is a sequential generation of the segments, which would mean that the segments that are generated earlier are not aware of the full context of the image. (iii) The content in textual prompts may sometimes be missed as the prompts for inpainting apply to the whole scene rather than a sub-scene generation.

9.2. Spatial fidelity

This is a key parameter for our evaluation. Ours:CD-TC scores highest followed by Ours:CD-T, B3:MD and B2:BLD.

Our take: This is on expected lines. Text-to-image (B1) provides no explicit control over the spatial layout apart from using natural language to describe the relative position of objects in a scene. It is also interesting to note that B1:TC does not score as high as Ours or B3, this strengthens our argument that using control conditions might not be sufficient to gain spatial control. The performance of B2:BLD can be attributed to the explicit use of a mask at each diffusion step to enforce spatial control. In B2:BLD the generated image is masked out using the inpainting mask and explicitly added to the base image. Similarly, B2:RSD is tuned for spatial control. Although B3:ediff-I manipulates the cross-attention mechanism to achieve finer spatial control, we notice that this is also somewhat limited.

9.3. Blending and harmony

This metric is hard to evaluate using automated metrics so we also borrow insights from the human surveys. Human-GP evaluation rates our method as the best, while Human-AD evaluation and automated methods give an edge to the serial inpainting method.

Our take: Text-to-Image (B1) generates one holistic image, and we expect it to produce a well-harmonized image. This higher rating for the serial-inpainting method could be due to the particular implementation of inpainting that we use in our project. This inpainting implementation (RunwayML SD 1.5 [40]) is especially fine-tuned to provide seamless filling of a masked region by direct inference similar to text-to-image generation. Further, in Composite Diffusion, the blending and harmonization are affected by the chosen scaffolding value, as shown in Supp table 4.

9.4. Technical quality

This metric is again hard to evaluate using automated metrics so we also borrow insights from the human surveys. Human evaluation-GP gives our method a better score, while Human evaluation-AP gives a slight edge to the other methods. The automated evaluation method considers only one aspect of technical quality, viz., the presence of noise; our algorithm shows fewer noise artifacts.

Our take: Both B2 and Ours build upon the base model B1:T. Any derivative approach risks losing the technical quality while attempting to introduce control. Hence, we expect the best-performing methods to maintain the technical quality displayed by B1. However, repeated application of inpainting to cover all the segments in B2 may amplify any noisy artifact introduced in the early stages. We also observed that for Composite Diffusion, if the segment masks do not have well-demarcated boundaries, we might get unwanted artifacts in the generated composites. B3:ediff-I on the other hand is similar to B1:T, except that it controls the cross-attention matrices associated with the different phrases in the input prompt to achieve spatial control.

9.5. Aesthetic quality

Ours:CD-TC gets the highest score followed by B1:TC and Ours:CD-T, B3:MD. B2 and B3:ediff-I get the lowest scores.

Our take: These results indicate that our approach does not cause any loss of aesthetic quality but may even enhance it. The good performance of Composite Diffusion in aesthetic evaluation can be due to the enhanced detail and nuance with both textual and spatial controls. Interestingly, the usage of control conditions enhances the aesthetic quality as evident in the results of Ours:CD-TC and B1:TC. The lack of global context of all the segments in serial inpainting and the dependence on an appropriate background image put it at a slight disadvantage. Aesthetics is a subjective criterion that can be positively influenced by having more meaningful generations and better placements of visual elements. Hence, combining segment layouts and content conditioning in Composite Diffusion may lead to compositions with more visually pleasing signals.

We further did a qualitative validation with an external artist. We requested the artist to specify her intent in the form of freehand drawings with labeled descriptions. We manually converted the artist's intent to bimodal input of segment layout and textual descriptions suitable for our model. We then created artwork through Composite Diffusion and asked the artist to evaluate them qualitatively. The feedback was largely positive and encouraging. The artist's inputs, the generated artwork, and the artist's feedback are available in the Supp section 10.

We also present a qualitative visual comparison of our generated outputs with the baselines and other related approaches in the main paper Fig. 5 and Supp figures 17,18, and 19 respectively. Summarizing the results of multiple modes of evaluation, we can affirm that our Composite Diffusion methods perform holistically and well across all the different quality criteria.

9.6. Scalability and computational efficiency

In this section, we discuss the computational efficiency of our method and the baselines that we compare against. We further discuss the scalability aspects of our approach

Computational efficiency: B1:T which only uses a single prompt as input takes the least amount of time to create an artwork. B1:TC which uses control conditions on top of it takes some additional time for a total of 7 seconds. Our:CD-T takes 13s which is in line with other related works like B3:MD, and similar to the additional time taken by B1:TC, Ours:CD-TC takes 19 seconds.

Our take: B1:T is on expected lines since it only takes a single input. B1:TC takes an additional 2s per prompt when compared to B1:T. This is also observed in Ours:CD-T which takes 13 seconds, and Ours:CD-TC takes an additional 2s per prompt (avg 3-4 prompts in the benchmark) to get a total of 19s to create an artwork. Ours:CD-T's creation time is comparable to related work such as B3:MD and B3:ediff-I. B2:RSD is specifically fine-tuned to take in as input masks and perform inpainting, and its diffusers implementation reads the mask within its inference method leading to some additional overhead in its reading and processing of masks. B2:BLD on the other hand uses B1:T as its base, however, its official implementation is particularly efficient considering that it only takes 7s to create an artwork, 2s more compared to B1:T.

Scalability In the domain of image generation, diffusion models have largely replaced GANs due to their powerful ability to generate high-fidelity images. This frontier of research is rapidly advancing, and larger and better diffusion models are continuously being developed. Within this rapidly evolving frontier Composite diffusion emerges as a flexible framework that is adaptable to emerging innovations in the diffusion model. This ensures that as state-of-the-art models emerge with new innovations or large sizes, composite diffusion can seamlessly integrate and scale with them.

One of the salient features of composite diffusion is its input versatility. We have already demonstrated how it can use diverse inputs such as lineart, scribble, human-pose, and reference images. This versatility expands the utility of composite diffusion and makes it a suitable choice for a wide array of potential applications.

Diffusion models do have one notable challenge, that is inference time. However, this is an active area of research in diffusion models and there have been recent works that aim to reduce inference time [36]. On the other hand, there has been massive engineering effort to improve the computational efficiency of existing diffusion models [15]. Some of these approaches include running model in half-precision weights, utilizing sliced and tiled VAE to combat low VRAM, utilizing memory efficient attention, etc. Using just a couple of these optimizations together results in 3.6x performance improvement [15]. Furthermore, the capability to process data in batches provides an avenue for efficient, large-scale operations.

10. Artworks exercise

Here, we present the outcomes of a brief collaboration with an artist. The aim of this exercise was to expose the modalities of our system to an external artist and gather early-stage user feedback. To achieve this, we explained the workings of the Composite Diffusion system to the artist and asked her to provide us with 2-3 specific scenarios of artwork that she would normally like to work on. The artist's inputs were given to us on plain sheets of paper in the form of rough drawings of the intended paintings, with clear labels for various objects and sections.

We converted these inputs into the bimodal input - the free-form segment layouts and text descriptions for each segment. We did not create any additional control inputs. We then supplied these inputs to our Composite Diffusion algorithm and performed many iterations with base and a few fine-tuned models, and also at different scaffolding values. The outputs were first shown to a few internal artists for quick feedback, and the final selected outputs were shared with the original artist. For the final shared outputs, refer to Figures 23 and 24 for input 1, Figures 25 and 26 for input 2, and Figures 27, 28, and 29 for input 3. Please note that the objective here was to produce an artwork with artistically satisfying outputs. So, for some of our generations, we even allowed the outputs which were creative and caught the overall intent of the artist, but did not strictly conform to the prescribed inputs by the artist.

The feedback that we received from the artist at the end of the exercise (as received on Jan 25, 2023), is presented here verbatim:

For artwork 1 (refer to Figures 23 and 24): *"The intended vision was of a lively scene with bright blue skies, picnic blossoms blooming, soft green grass with fallen pink petals, and a happy meal picnic basket. All the images are close enough to the description. Colors are bright and objects fit harmoniously with each other."*

For artwork 2 (refer to Figures 25 and 26): *"The intended vision was of bears in their natural habitat, surrounded by forest trees and snow-clad mountains, catching fish in the stream. Overall, the bears, mountains, trees, rocks, and streams are quite realistic. However, not a single bear could catch a fish. Bear 4 looks like an Afgan hound (dog breed with long hair) and bear 5 itself became a mountain. In image 10, the objects have merged into one another, having ill-defined margins and boundaries."*

For artwork 3(refer to Figures 27, 28, and 29): *"The intended vision was of an angel - symbolic of hope & light, salvaging a dejected man, bound by the shackles of hopelessness and despair. Each and every angel is a paragon of beauty. Compared to the heavenly angels, the desolate men look a bit alienish. There is a slight disproportion seen between man and angel in some images. My personal best ones are no. 1,3,6,10."*

In another case, we wanted to check if our system can be effectively used for artistic pattern generation. Here we gave the system an abstract pattern in the form of a segment layout and specified the objects that we want to fill within those segments. Figure 30 shows a sample output of such an exercise where we fill in the pattern with different forms of flowers.

While the exercise of interactions with the artist and the application of the system for creating the artwork was informal and done in a limited manner, still, it demonstrated to us at an early stage the effectiveness of Composite Diffusion in creating usable art in real-life scenarios. It also validated that the art workflow was simple and intuitive enough to be adopted by people with varied levels of art skills.

References

[1] Andrew. Stable diffusion samplers: A comprehensive guide, June 2023. 2

[2] AQ. Finetuned diffusion - a hugging face space. https://huggingface.co/spaces/anzorq/finetuned_diffusion, 2022. 8

[3] Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion, 2022. 5, 19, 21, 22

[4] 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, June 2023. 19, 20, 21, 28

[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. 5, 19

[6] 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 an ensemble of expert denoisers, 2023. 19, 20, 23

[7] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. *arXiv preprint arXiv:2302.08113*, 2023. 19, 20, 21, 23

[8] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 417–424, 2000. 5

[9] Ali Borji. Pros and cons of gan evaluation measures: New developments. *Computer Vision and Image Understanding*, 215:103329, 2022. 27

[10] Zoya Bylinskii, Laura Herman, Aaron Hertzmann, Stefanie Hutka, and Yile Zhang. Towards better user studies in computer graphics and vision. *arXiv preprint arXiv:2206.11461*, 2022. 26

[11] Guangyong Chen, Fengyuan Zhu, and Pheng Ann Heng. An efficient statistical method for image noise level estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 477–485, 2015. 27

[12] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022. 19

[13] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021. 2, 3, 19

[14] Sander Dieleman. Guidance: a cheat code for diffusion models, 2022. 1, 2, 3

[15] Diffusers. Optimization fp16 documentation, 2023. Accessed: 1st Sept 2023. 30

[16] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. 2020. 19

[17] 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. 19, 21

[18] Federico Galatolo., Mario Cimino., and Gigliola Vaglini. Generating images from caption and vice versa via clip-guided generative latent space search. *Proceedings of the International Conference on Image Processing and Vision Engineering*, 2021. 3

[19] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. 2022. 19

[20] 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. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 27

[21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1, 2

[22] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 3, 4

[23] Xun Huang, Arun Mallya, Ting-Chun Wang, and Ming-Yu Liu. Multimodal conditional image synthesis with product-of-experts gans, 2021. 19

[24] LAION-AI. aesthetic-predictor. https://github.com/LAION-AI/aesthetic-predictor, 2022. 27, 28

[25] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. *arXiv preprint arXiv:2206.01714*, 2022. 19

[26] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B. Tenenbaum. Compositional visual generation with composable diffusion models, 2022. 19

[27] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11461–11471, June 2022. 5, 19

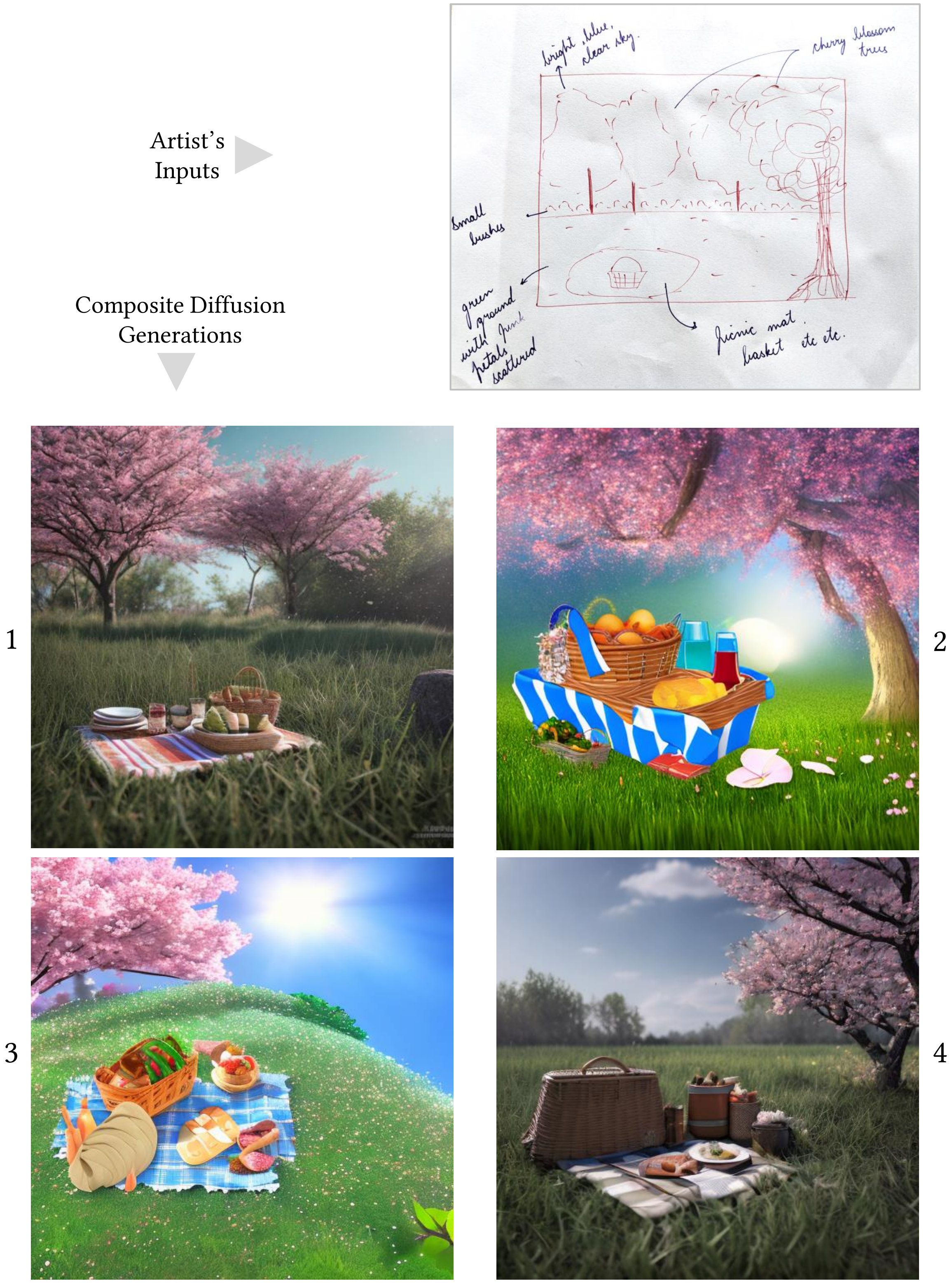


Figure 23. Artwork Exhibit 1a

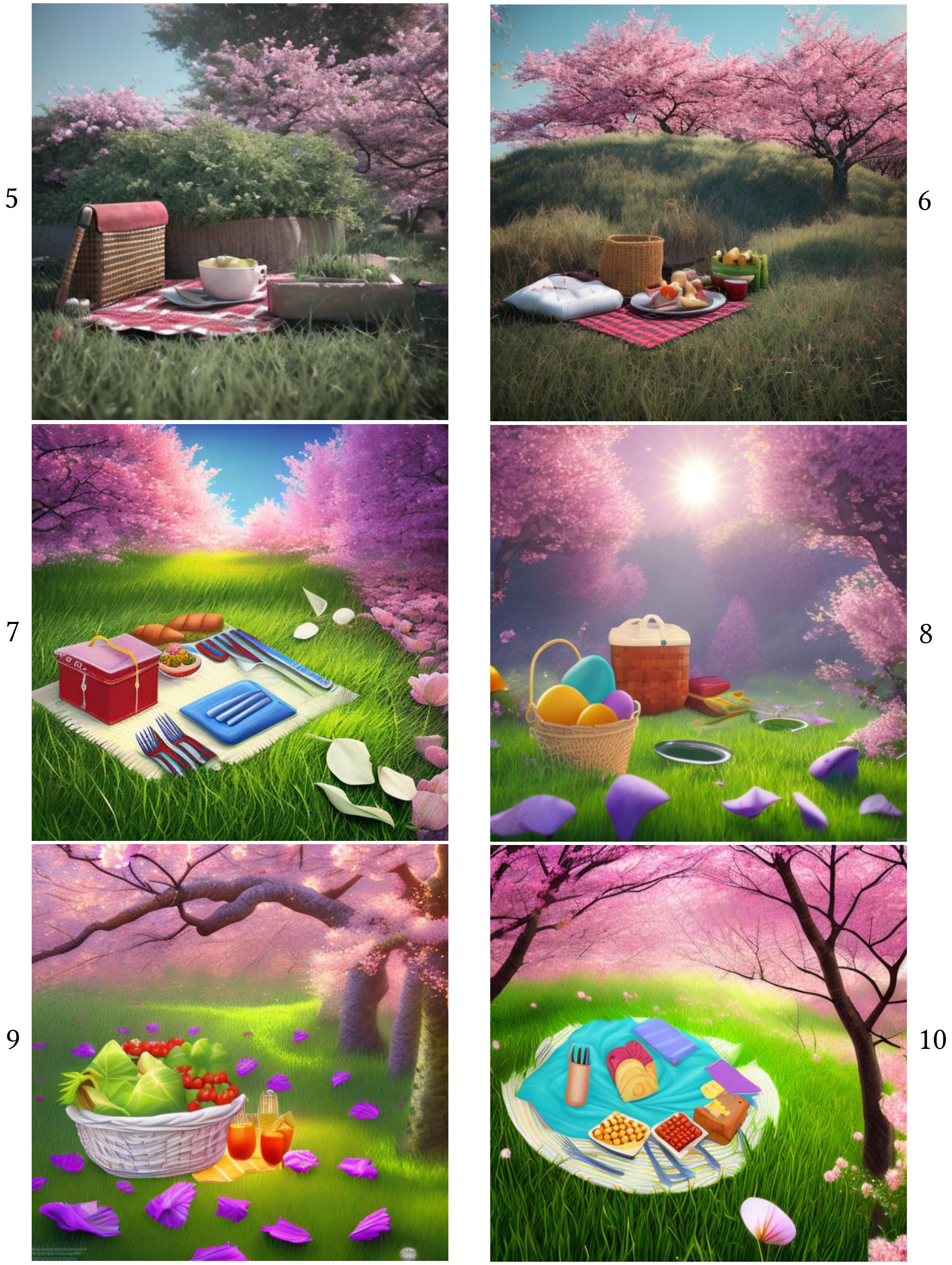


Figure 24. Artwork Exhibit 1b

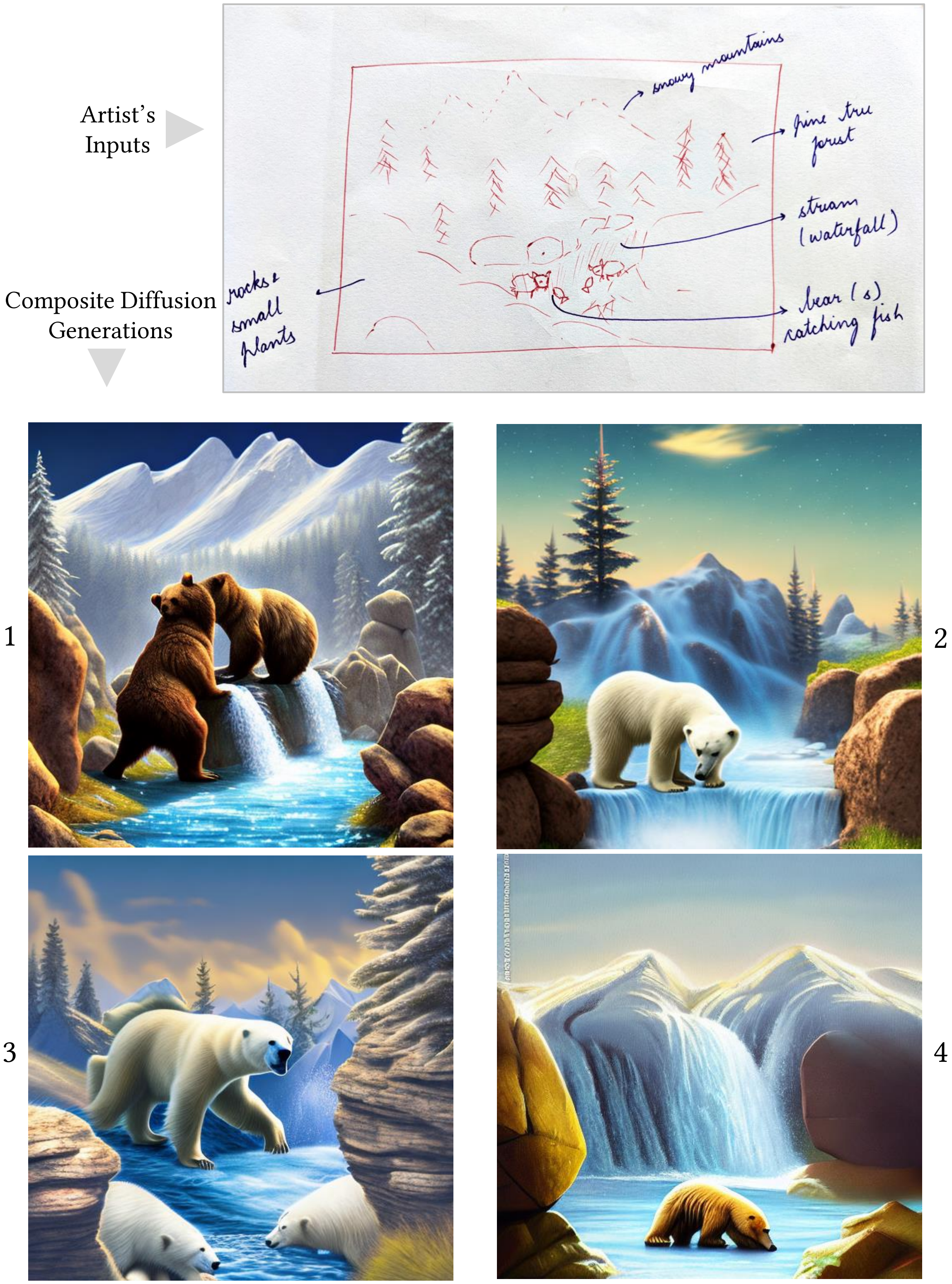


Figure 25. Artwork Exhibit 2a

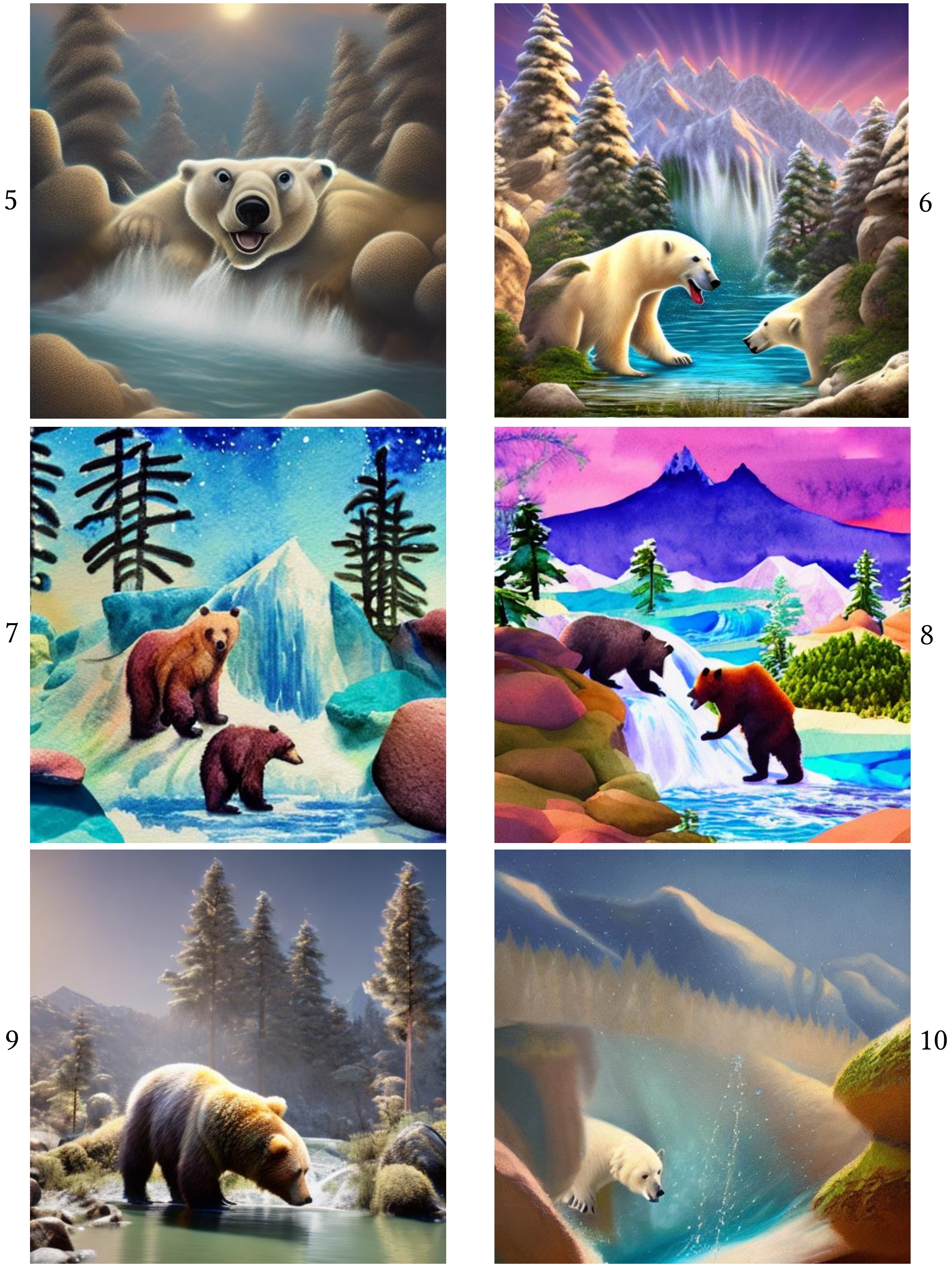


Figure 26. Artwork Exhibit 2b

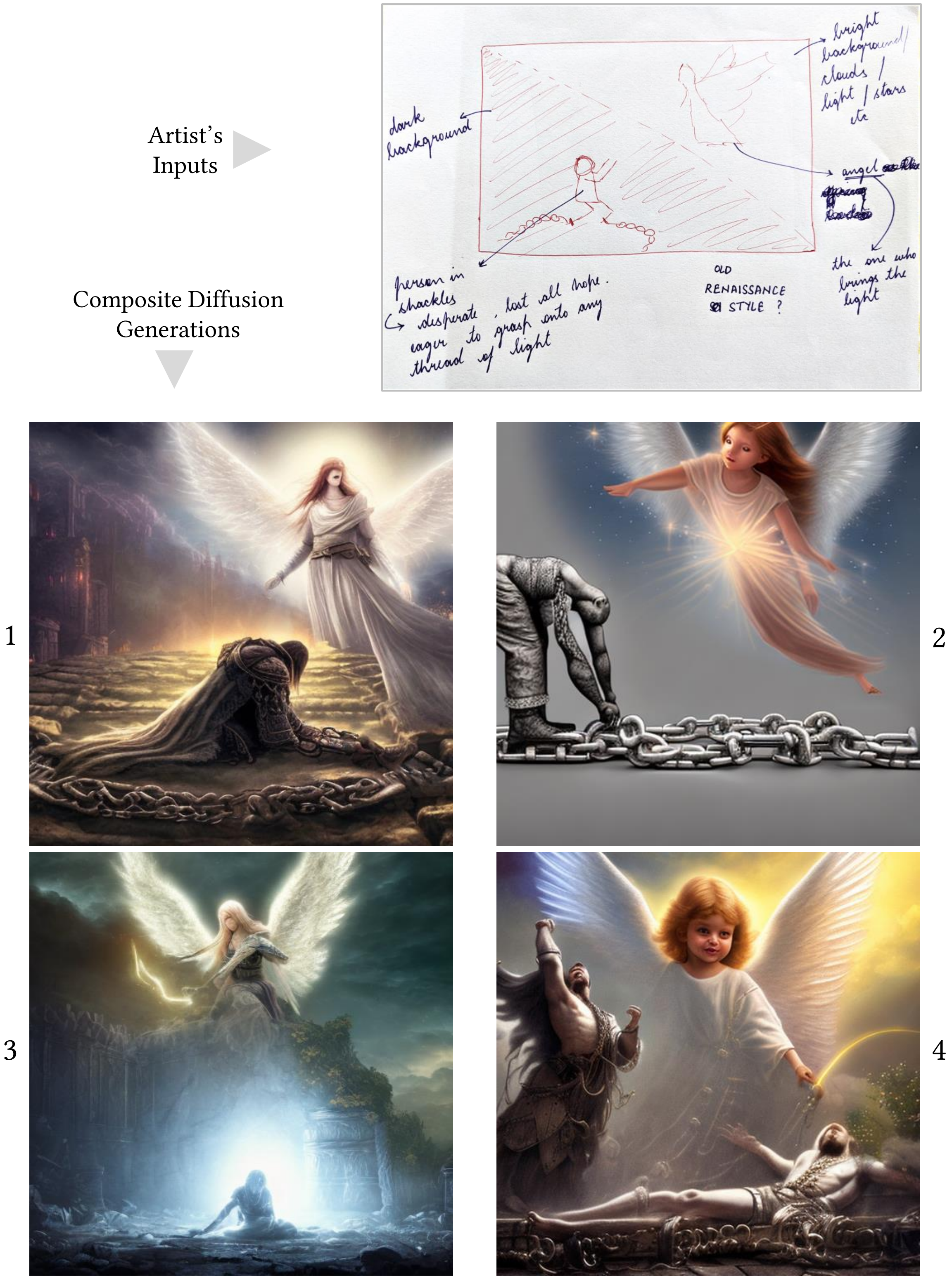

Artist's Inputs
Composite Diffusion Generations
dark background
bright background / clouds / light / stars etc
angel
the one who brings the light
person in shackles
desperate, lost all hope.
eager to grasp onto any thread of light
OLD RENAISSANCE STYLE ?
1
2
3
4

Figure 27. Artwork Exhibit 3a

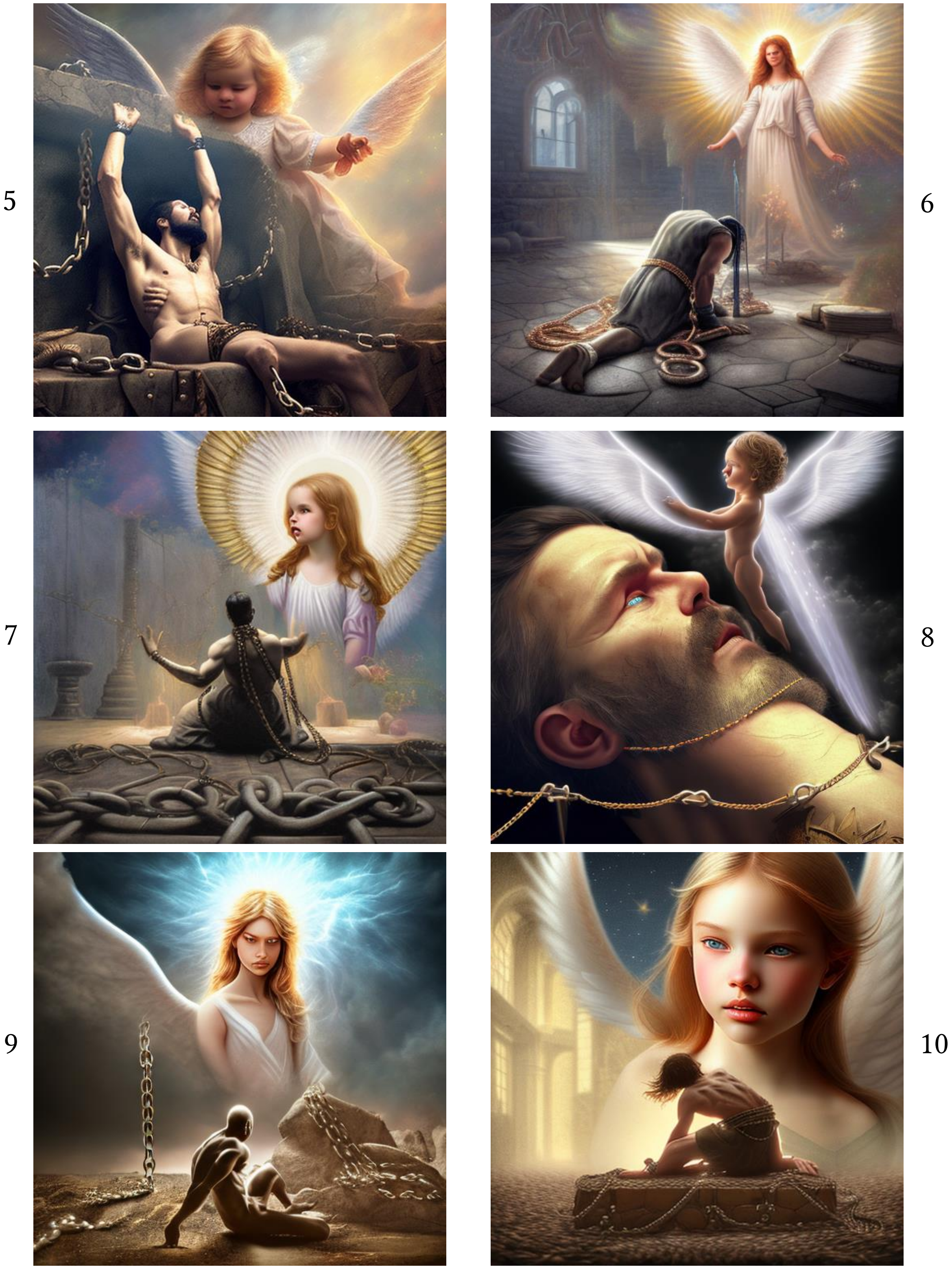


Figure 28. Artwork Exhibit 3b

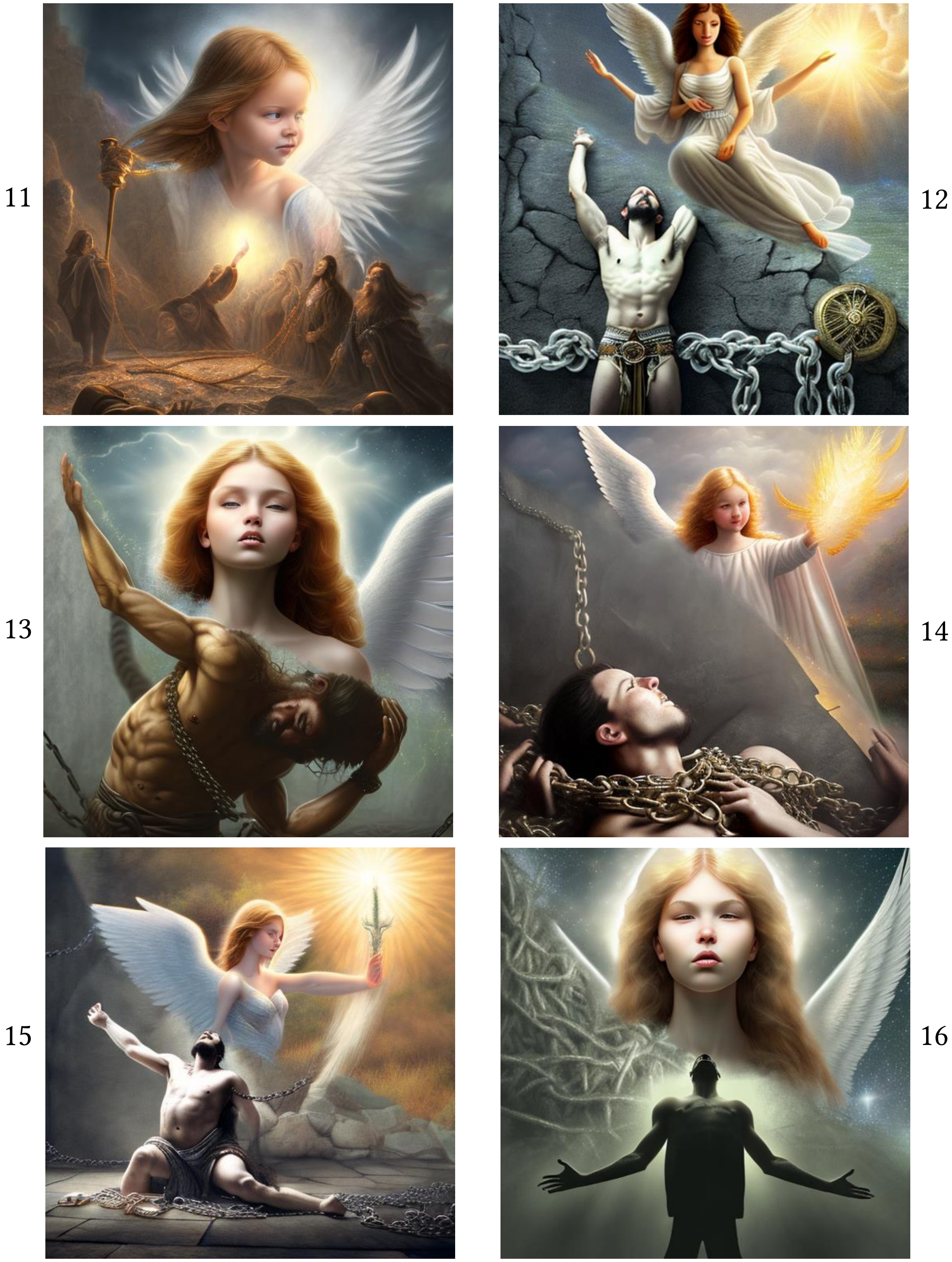


Figure 29. Artwork Exhibit 3c

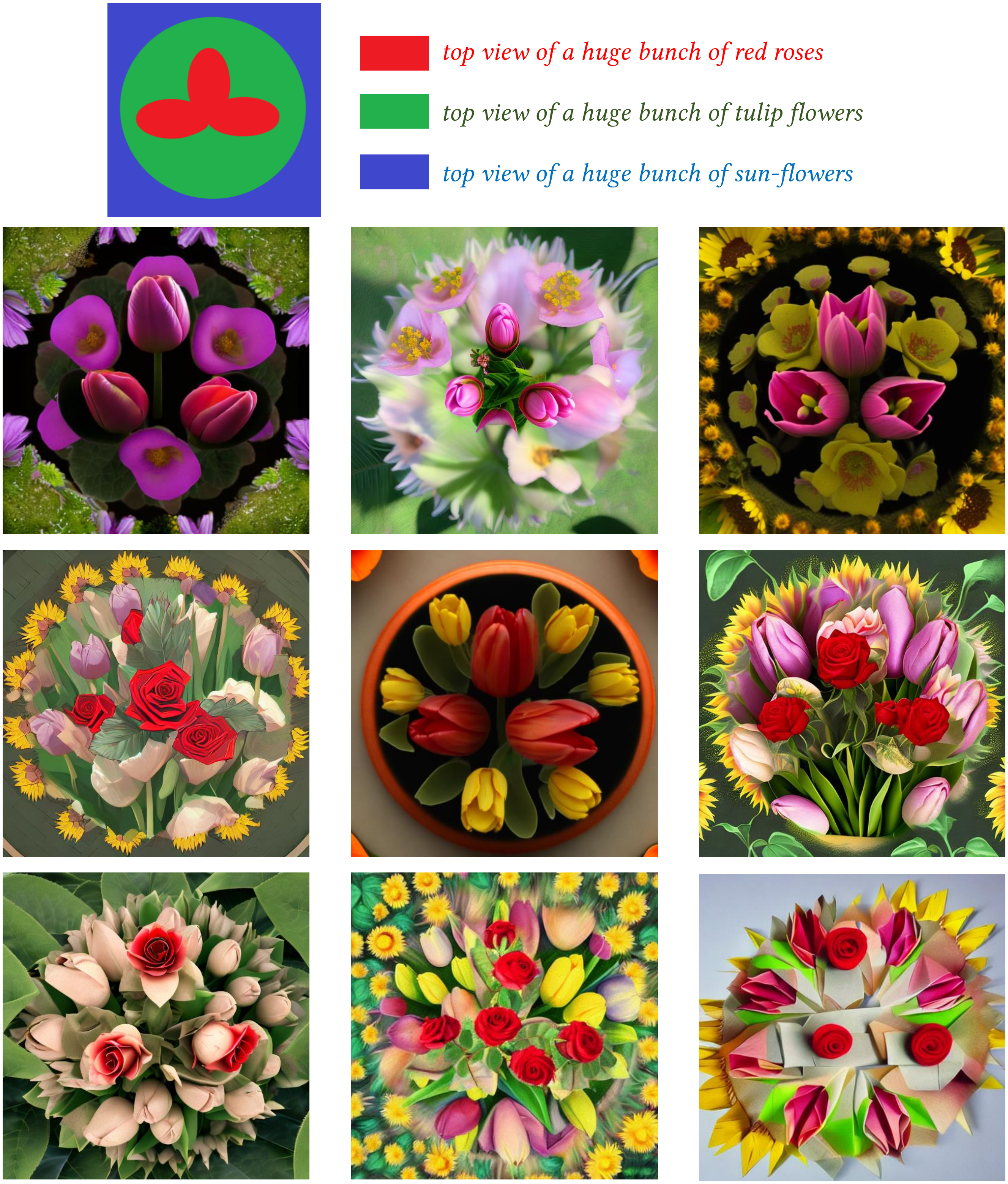

🟥 top view of a huge bunch of red roses
🟩 top view of a huge bunch of tulip flowers
🟦 top view of a huge bunch of sun-flowers

Figure 30. Artwork Exhibit 4: Flower Patterns

[28] Agata Mlynarczyk. Stable diffusion and the samplers mystery, March 2023. 2

[29] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. *arXiv preprint arXiv:2211.09794*, 2022. 19

[30] 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. 2021. 3, 19

[31] OpenAI. Guided diffusion. `https://github.com/openai/guided-diffusion`, 2021. 19

[32] Roni Paiss, Hila Chefer, and Lior Wolf. No token left behind: Explainability-aided image classification and generation, 2022. 19

[33] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. 2019. 19

[34] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. 2021. 3

[35] Ford Paul. Dear artists: Do not fear ai image generators, 2022. 12

[36] William Peebles and Saining Xie. Scalable diffusion models with transformers, 2023. 30

[37] 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. 2021. 3, 28

[38] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022. 19

[39] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. 2021. 19

[40] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. `https://github.com/runwayml/stable-diffusion`, 2021. 2, 4, 6, 9, 19, 22, 29

[41] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. 2

[42] 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. 2022. 19

[43] 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, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022. 19

[44] Mehdi SM Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. Assessing generative models via precision and recall. *Advances in neural information processing systems*, 31, 2018. 27

[45] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training gans. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. 27

[46] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2, 4

[47] Lilian Weng. What are diffusion models? *lilianweng.github.io*, Jul 2021. 1

[48] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation, 2023. 27, 28

[49] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. `https://github.com/lllyasviel/ControlNet-v1-1-nightly`, 2023. 7, 22