High-Fidelity Guided Image Synthesis with Latent Diffusion Models

Jaskirat Singh[1] Stephen Gould[1,2] Liang Zheng[1,2]
[1]The Australian National University [2]Australian Centre for Robotic Vision
{jaskirat.singh, stephen.gould, liang.zheng}@anu.edu.au

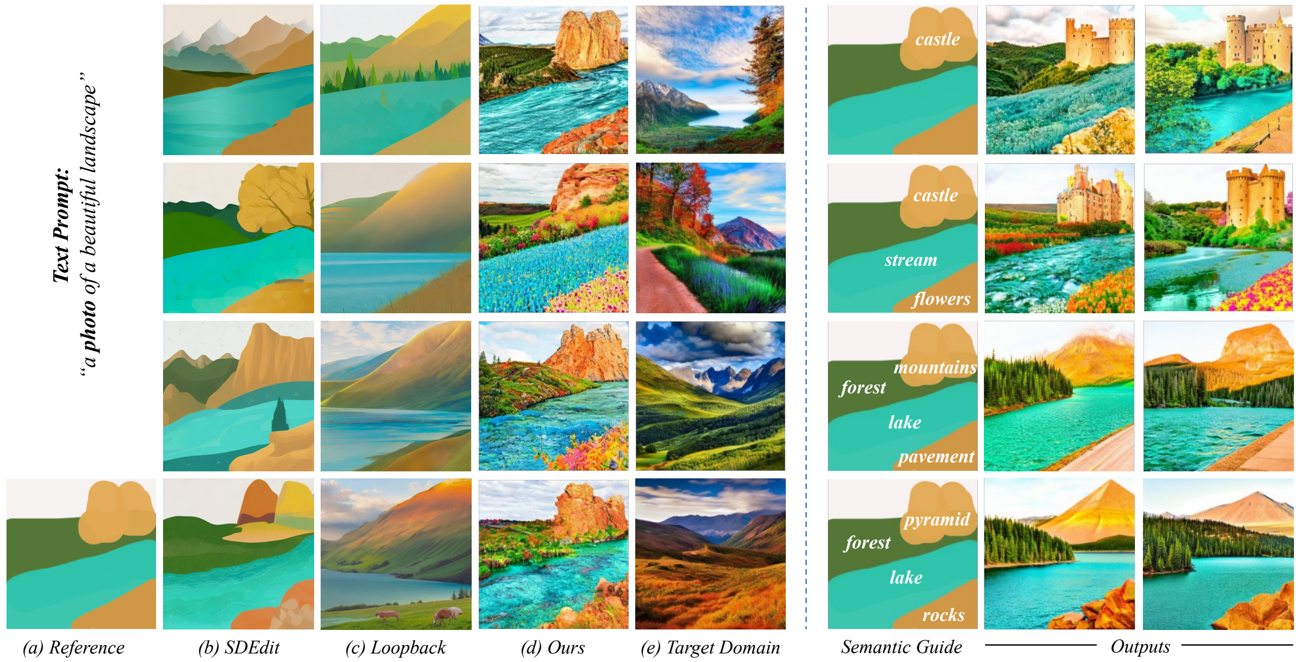


Figure 1. ***Overview***. We propose a novel stroke based guided image synthesis framework which *(Left)* resolves the intrinsic domain shift problem in prior works (b), wherein the final images lack details and often resemble simplistic representations of the target domain (e) (generated using only text-conditioning). Iteratively reperforming the guided synthesis with the generated outputs (c) seems to improve realism but it is expensive and the generated outputs might lose faithfulness with the reference (a) with each iteration. *(Right)* Additionally, the user is also able to specify the semantics of different painted regions without requiring conditional training or finetuning.

Abstract

Controllable image synthesis with user scribbles has gained huge public interest with the recent advent of text-conditioned latent diffusion models. The user scribbles control the color composition while the text prompt provides control over the overall image semantics. However, we note that prior works suffer from an intrinsic domain shift problem wherein the generated outputs often lack details and resemble simplistic representations of the target domain. In this paper, we propose a novel guided image synthesis framework, which addresses this problem by modelling the output image as the solution of a constrained optimization problem. We show that while computing an exact solution to the optimization is infeasible, an approximation of the same can be achieved while just requiring a single pass of the reverse diffusion process. Additionally, we show that by simply defining a cross-attention based correspondence between the input text tokens and the user stroke-painting, the user is also able to control the semantics of different painted regions without requiring any conditional training or finetuning. Human user study results show that the proposed approach outperforms the previous state-of-the-art by over 85.32% on the overall user satisfaction scores. Source code and demo are available at our project page https://1jsingh.github.io/gradop.

1. Introduction

Guided image synthesis with user scribbles has gained widespread public attention with the recent advent of large-scale language-image (LLI) models [25, 28, 30, 32, 42]. A novice user can gain significant control over the final image contents by combining text-based conditioning with unsupervised guidance from a reference image (usually a coarse stroke painting). The text prompt controls the overall image semantics, while the provided coarse stroke painting allows the user to define the color composition in the output scene.

Existing methods often aim attempt to achieve this through two means. The first category leverages conditional training using semantic segmentation maps [8, 30, 41]. However, the conditional training itself is quite time-consuming and requires a large scale collection of dense semantic segmentation labels across diverse data modalities. The second category, typically leverages an inversion based approach for mapping the input stroke painting to the target data manifold without requiring any paired annotations. For instance, a popular solution by [16, 23, 37] introduces the painting based generative prior by considering a noisy version of the original image as the start of the reverse diffusion process. However, the use of an inversion based approach causes an intrinsic domain shift problem if the domain gap between the provided stroke painting and the target domain is too high. In particular, we observe that the resulting outputs often lack details and resemble simplistic representations of the target domain. For instance, in Fig. 1, we notice that while the target domain consists of *realistic photos* of a landscape, the generated outputs resemble simple pictorial arts which are not very realistic. Iteratively reperforming the guided synthesis with the generated outputs [4] seems to improve realism but it is costly, some blurry details still persist (refer Fig. 4), and the generated outputs tend to lose faithfulness to the reference with each successive iteration.

To address this, we propose a diffusion-based guided image synthesis framework which models the output image as the solution of a constrained optimization problem (Sec. 3). Given a reference painting y, the constrained optimization is posed so as to find a solution x with two constraints: 1) upon painting x with an autonomous painting function we should recover a painting similar to reference y, and, 2) the output x should lie in the target data subspace defined by the text prompt (*i.e.*, if the prompt says *"photo"* then we want the output images to be realistic photos instead of cartoon-like representations of the same concept). Subsequently, we show that while the computation of an exact solution for this optimization is infeasible, a practical approximation of the same can be achieved through simple gradient descent.

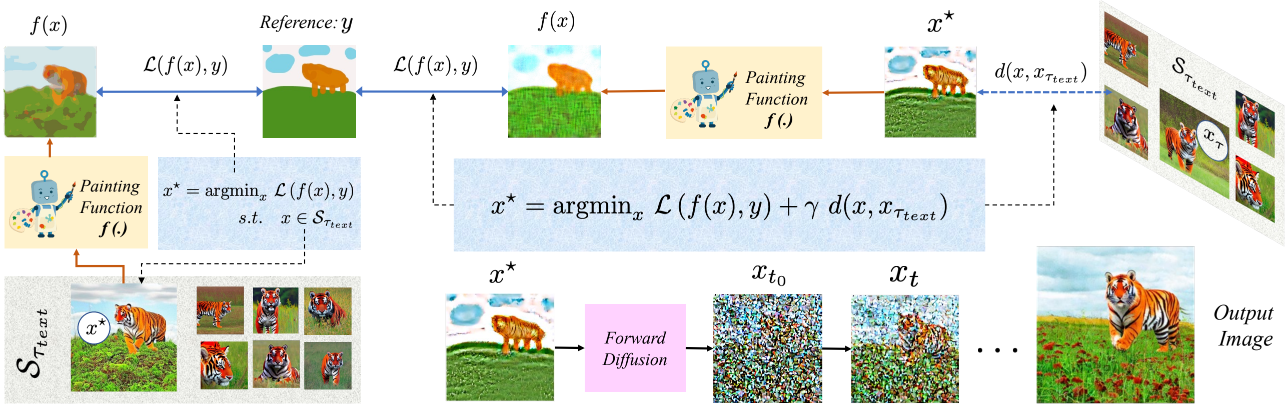
Finally, while the proposed optimization allows the user to generate image outputs with high realism and faithfulness (with reference y), the fine-grain semantics of different painting regions are inferred implicitly by the diffusion model. Such inference is typically dependent on the generative priors learned by the diffusion model, and might not accurately reflect the user's intent in drawing a particular region. For instance, in Fig. 1, we see that the light blue regions can be inferred as blue-green grass instead of a river. To address this, we show that by simply defining a cross-attention based correspondence between the input text tokens and user stroke-painting, the user can control semantics of different painted regions without requiring semantic-segmentation based conditional training or finetuning.

2. Related Work

GAN-based methods have been extensively explored for performing guided image synthesis from coarse user scribbles. [1–3, 15, 29, 39, 44] use GAN-inversion for projecting user scribbles on to manifold of real images. While good for performing small scale inferences these methods fail to generate highly photorealistic outputs when the given stroke image is too far from the real image manifold. Conditional GANs [7,14,19,22,26,38,45] learn to directly generate realistic outputs based on user-editable semantic segmentation maps. In another work, Singh *et al.* [36] propose an image synthesis framework which leverages autonomous painting agents [20, 34, 35, 46] for inferring photorealistic outputs from rudimentary user scribbles. Despite its efficacy, this requires the creation of a new dataset and conditional training for each target domain, which is expensive.

Guided image synthesis with LLI models [6,25,28,30, 32,42,43] has gained widespread attention [9,13,16,18,21, 31, 33] due to their ability to perform high quality image generation from diverse target modalities. Of particular interest are works wherein the guidance is provided using a coarse stroke painting and the model learns to generate outputs conditioned on both text and painting. Current works in this direction typically 1) use semantic segmentation based conditional training [8, 30, 41] which is expensive, or, 2) adopt an inversion-based approach for mapping the input stroke painting to the target data manifold without requiring paired annotations. For instance, Meng *et al.* [16, 23] propose guided image synthesis framework, wherein the generative prior is introduced by simply considering a noisy version of the original sketch input as the start of the reverse diffusion process. Choi *et al.* [5] propose an iterative conditioning strategy wherein the intermediate diffusion outputs are successively refined to move towards the reference image. While effective, the use of an inversion-like approach causes an implicit domain shift problem, wherein the output images though faithful to the provided reference show blurry or less textured details. Iteratively reperforming guided synthesis with generated outputs [4] seems to improve realism but it is costly. In contrast, we show that it is possible to perform highly photorealistic image synthesis while just requiring a single reverse diffusion pass.

Cross-attention control. Recently, Hertz *et al.* [10] propose a prompt-to-prompt image editing approach with text-conditioned diffusion models. By constraining the cross-attention features of all non-targeted text tokens to remain


$f(x)$
$\mathcal{L}(f(x),y)$
Reference: y
$\mathcal{L}(f(x),y)$
$f(x)$
Painting Function $f(.)$
$x^\star$
$d(x, x_{\tau_{text}})$
$\mathcal{S}_{\tau_{text}}$
x_τ
Painting Function $f(.)$
$x^\star = \text{argmin}_x\ \mathcal{L}(f(x),y)$ s.t. $x \in \mathcal{S}_{\tau_{text}}$
$x^\star = \text{argmin}_x\ \mathcal{L}(f(x),y) + \gamma\ d(x, x_{\tau_{text}})$
$\mathcal{S}_{\tau_{text}}$
$x^\star$
$x^\star$
Forward Diffusion
x_{t_0}
x_t
Output Image

(a) Constrained Optimization Formulation

(b) Obtaining an Approximate Solution

Figure 2. ***Method Overview.*** (a) Given a reference painting y and text prompt τ_{text}, we first formulate the guided synthesis problem as the solution $x^\star$ of a constrained optimization problem with 2 properties: 1) $x^\star$ lies in the subspace $\mathcal{S}_{\tau_{text}}$ of outputs conditioned only on the text, and, 2) upon painting x we should recover reference painting y. While computing an exact solution of this optimization is infeasible, we show that an approximation can be obtained by solving the unconstrained optimization in (b). Here we first use gradient descent to compute a point $x^\star$ close to a random sample $x_{\tau_{text}} \in \mathcal{S}_{\tau_{text}}$, while still minimizing the painting loss $\mathcal{L}(f(x), y)$. This $x^\star$ is usually non-photorealistic due to gradient descent. We therefore use the diffusion based inversion from [37] to map it back to target domain $\mathcal{S}_{\tau_{text}}$.

the same, they show that by only modifying the text prompt, it is possible to perform diverse image editing without changing the underlying structure of the original input image. In contrast, we use cross-attention control for from-scratch synthesis and show that by simply defining a cross-attention based correspondence between input text tokens and the user stroke-painting, it is possible to control and define the fine-grain semantics of different painted regions.

3. Our Method

Let $f : \mathcal{D}_{real} \rightarrow \mathcal{D}_{paint}$ be a function mapping a real input image x to its painted image $f(x)$. Then given a colored stroke image y and input text prompt τ_{text}, we formulate the computation of output image $x^\star$ as the solution to the following constrained optimization problem,

$$x^\star = \text{argmin}_x\ \mathcal{L}\left(f(x), y\right) \tag{1}$$

$$\text{subject to } x \in \mathcal{S}_{\tau_{text}} \tag{2}$$

where $\mathcal{L}\left(f(x), y\right)$ represents a distance measure between the painted output $f(x)$ of image x and the target painting y, while $\mathcal{S}_{\tau_{text}}$ represents the subspace of output images conditioned only on the text input.

In other words, by additionally conditioning on a stroke image y, we wish to find a solution $x^\star$ such that 1) the distance between the painted image of x and reference painting y is minimized, while at the same time ensuring 2) the final solution lies in the subspace of images conditioned only on the text prompt τ_{text}. For instance, if the text says *"a realistic photo of a tree"* then the use of stroke-based guidance should still produce a *"realistic photo"*, wherein the composition of the tree regions is controlled by the painting y.

3.1. GradOP: Obtaining an Approximate Solution

The optimization problem in Eq. 1 can be reformulated as an unconstrained optimization problem as,

$$x^\star = \text{argmin}_x\ \mathcal{L}\left(f(x), y\right) + \gamma\ d(x, \mathcal{S}_{\tau_{text}}), \tag{3}$$

where $d(x, \mathcal{S}_{\tau_{text}})$ represents a distance measure between x and subspace $\mathcal{S}_{\tau_{text}}$, and γ is a hyperparameter.

A cursory glance at the above formulation should make it evident that the computation of an exact solution is infeasible without first generating a large enough sample size for the $\mathcal{S}_{\tau_{text}}$ subspace, which will be quite time consuming.

To address this, we propose to obtain an approximate solution by estimating $d(x, \mathcal{S}_{\tau_{text}})$ through the distance of x from a single random sample $x_{\tau_{text}} \in \mathcal{S}_{\tau_{text}}$. Thus, we can approximate the optimization problem as follows,

$$x^\star = \text{argmin}_x\ \mathcal{L}\left(f(x), y\right) + \gamma\ d(x, x_{\tau_{text}}). \tag{4}$$

Assuming a latent diffusion model with decoder $\mathcal{D}$, we can rewrite the above above optimization in latent space as,

$$z^\star = \text{argmin}_z\ \mathcal{L}\left(f(\mathcal{D}(z)), y\right) + \gamma\ \|z - z_{\tau_{text}}\|_2. \tag{5}$$

where the image output $x^\star$ can be computed as $x^\star = \mathcal{D}(z^\star)$.

In order to solve the above optimization problem, we first use the diffusion model to sample $x_{\tau_{text}} \in \mathcal{S}_{\tau_{text}}$. Initializing $z = \mathcal{E}(x_{\tau_{text}})$, where $\mathcal{E}$ represents the encoder, we solve the above optimization using gradient descent (assuming f and $\mathcal{L}$ are differentiable). Finally, we note that the solution $x^\star = \mathcal{D}(z^\star)$ to the above approximation of the optimization problem might be non-photorealistic due to gradient descent. We therefore use the diffusion based inversion approach from [16] in order to map it to the target image manifold. Please refer Alg. 1 for the detailed implementation.

Algorithm 1 GradOP: Solution Approximation

Input: Stroke Painting y, text prompt τ_{text}
Output: Output image x conditioned on both τ_{text}, y
Require: Differentiable painting function f, differentiable distance measure $\mathcal{L}$, hyperparameter γ, t_0.

```
Sample $x_{\tau_{text}} \in \mathcal{S}_{\tau_{text}}$;
Initialize $z = z_{\tau_{text}} = \mathcal{E}(x_{\tau_{text}})$;
for $0 \leq i \leq M$ do
    $\mathcal{L}_{total} = \mathcal{L}\left(f(\mathcal{D}(z)), y\right) + \gamma \ \|z - z_{\tau_{text}}\|_2$;
    $z = z - \lambda \nabla_z \mathcal{L}_{total}$;
end for
$z_{t_0} = \textsc{ForwardDiff}(z^\star = z, 0 \rightarrow t_0)$;
$z = \textsc{ReverseDiff}(z_{t_0}, t_0 \rightarrow 0)$;
return $x_{out} = \mathcal{D}(z)$.
```

3.2. GradOP+ : Improving Sampling Efficiency

While the guided synthesis solution in Alg. 1 shows great output results, for each output image it first requires the sampling of a text-only conditioned output $x_{\tau_{text}} \in \mathcal{S}_{\tau_{text}}$. To address this, we propose a modified guided image synthesis approach which allows for equally high quality outputs while requiring just a single reverse diffusion pass for each output. Our key insight is that a lot of information in $z^\star$ is discarded during the forward diffusion from $z^\star \rightarrow z_{t_0}$. Thus, instead of performing the optimization to first compute $z^\star$, we would like to directly optimize the intermediate latent states z_t by injecting the optimization gradients within the reverse diffusion process itself (refer Fig. 3).

In particular, at any timestep t during the reverse diffusion, we wish to introduce optimization gradients in order to solve the following optimization problem,

$$z_t^\star = \text{argmin}_z \ \mathcal{L}\left(f(\mathcal{D}(z)), y\right) + \gamma \ \|z - z_t\|_2. \qquad (6)$$

However, the introduction of gradients will cause $z_t^\star$ to not conform with the expected latent distribution at timestep t. We therefore pass it through the forward diffusion process in order to map it back to the expected latent variable distribution. Please refer Alg. 2 for the detailed implementation.

3.3. Controlling Semantics of Painted Regions

Finally, while the above approximate guided image synthesis algorithm allows for generation of image outputs with high *faithfulness* and *realism*, the semantics of different painted regions are inferred in an implicit manner. Such inference is typically based on the cross-attention priors (learned by the diffusion model) between the provided text tokens and the input painting throughout the reverse diffusion process. For instance, in the first example from Fig. 5, we note that for different outputs, the blue region can be inferred as a river, waterfall, or a valley. Also note that some painting regions might be entirely omitted (*e.g.* the brown strokes for the hut), if the model does not understand that the corresponding strokes indicate a distinct semantic entity *e.g.* a hut, small castle *etc*. Moreover, as shown in Fig. 5 such discrepancies persist even if the corresponding text tokens (*e.g.* a hut) are added to the textual prompt.

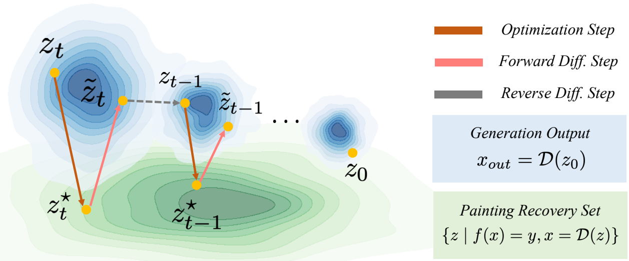


Figure 3. ***GradOP+ Overview.*** At any timestep t, the optimization in Eq. 6 ($z_t \rightarrow z_t^\star$) reduces the painting recovery loss, while the forward diffusion step $z_t^\star \rightarrow \tilde{z}_t$ maps it back to the expected latent distribution. By iteratively performing this optimization, *GradOP+* modifies the reverse sampling trajectory to lead to output $x_{out} = \mathcal{D}(z_0)$ which is also faithful to the target painting y.

Algorithm 2 GradOP+ : Improving Sampling Efficiency

Input: Stroke Painting y, text prompt τ_{text}
Output: Output image x conditioned on both τ_{text}, y
Require: Differentiable painting function f, distance measure $\mathcal{L}$, hyperparameter $\gamma, t_0, t_{start}, t_{end}$.

```
Sample $z_T \sim \mathcal{N}(0, \boldsymbol{I})$;
for $0 \leq t < T$ do
    $z_t = \textsc{ReverseDiff}(z_{t+1}, t+1 \rightarrow t)$;
    if $t_{start} \leq t \leq t_{end}$ then
        Initialize $z = z_t$;
        for $0 \leq i \leq M$ do
            $\mathcal{L}_{total} = \mathcal{L}\left(f(\mathcal{D}(z)), y\right) + \gamma \ \|z - z_t\|_2$;
            $z = z - \lambda \nabla_z \mathcal{L}_{total}$;
        end for
        $z_t = \textsc{ForwardDiff}(z_t^\star = z, 0 \rightarrow t)$
    end if
end for
return $x_{out} = \mathcal{D}(z_0)$.
```

Our key motivation is that the when generated faithfully, the average attention maps across different cross-attention layers show high overlap with the target object segmentation during the initial to intermediate parts of the reverse diffusion process. In our experiments, we found the reverse to also be true. That is, by constraining the cross attention map corresponding to a target semantic label to have a high overlap with the desired painting region, it is possible to control the semantics of different painting regions without the need for segmentation based conditional training.

In particular, given the binary masks corresponding to different painting regions $\{\mathcal{B}_1, \dots \mathcal{B}_N\}$ and the corresponding semantic labels $\{u_1, \dots u_N\}$, we first modify the input text tokens as follows,

$$\tau_{modified} = \tau + \{\text{CLIP}(u_i) \mid i \in [1, N]\}\,, \qquad (7)$$

where τ is the set of CLIP [27] tokens for input text prompt.

At any timestep $t \in [0, T]$ during the reverse diffusion process, we then enforce semantic control by modifying the cross-attention map $\mathcal{A}_t^i$ corresponding to label u_i as follows,

$$\tilde{\mathcal{A}_t^i} = w_i \left[(1 - \kappa_t)\, \mathcal{A}_t^i + \kappa_t \, \frac{\mathcal{B}_i}{\|\mathcal{B}_i\|_F} \, \|\mathcal{A}_t^i\|_F \right] \qquad (8)$$

where $\|.\|_F$ represents the Frobenius norm, $\kappa_t = t/T \in [0, 1]$ helps regulate the overlap between the cross-attention output $\tilde{\mathcal{A}_t^i}$ and the desired painting region $\mathcal{B}_i$ during the reverse diffusion process, and, weights $w_i,\ i \in [1, N]$ help the user to control the relative importance of expressing different semantic concepts in the final image.

4. Experiments

Implementation Details. We use publicly available text-conditioned latent diffusion models [30, 40] for implementing the purposed approach in Sec. 3. The constrained optimization is performed using gradient descent with the Adam [17] optimizer and number of gradient steps $N_{grad} \in [20, 60]$ (please refer Sec. 5.2 for detailed analysis). While several formulations of the distance measure $\mathcal{L}$ and painting function f are possible (refer supp. material for details), we find that simply approximating the function $\mathcal{L}$ using mean squared distance and f as a convolution operation with a gaussian kernel seems to give the fastest inference time performance with our method. For consistency reasons, we use the non-differentiable painting function from SDEdit [23] while reporting quantitative results (refer Sec. 4.1).

4.1. Stroke Guided Image Synthesis

Evaluation metrics. Given an input stroke painting, we compare the performance of our approach with prior works in guided image synthesis when no paired data is available. The performance of the final outputs is measured in terms of both *faithfulness* of the generated image with the target stroke painting as well as the *realism* of the final output distribution. In particular, given an input painting y and output real image prediction x, we define faithfulness $\mathcal{F}(x, y)$ as,

$$\mathcal{F}(x, y) = \mathcal{L}_2(f(x), y) \qquad (9)$$

where $f(.)$ is the painting function. Thus an output image x is said to have high faithfulness with the given painting y if upon painting the final output x we get a painting $\tilde{y} = f(x)$ which is similar to the original target painting y.

Similarly, given a set of output data samples $\mathcal{S}(y, \tau_{text})$ conditioned on both painting y and text τ_{text}, and, $\mathcal{S}(\tau_{text})$ conditioned only on the text, the *realism* $\mathcal{R}$ is defined as,

$$\mathcal{R}(\mathcal{S}(y, \tau_{text})) = FID\,(\mathcal{S}(y, \tau_{text}), \mathcal{S}(\tau_{text})) \qquad (10)$$

where FID represents the Fisher inception distance [11].

Method	Evaluation criteria		User Study Results	
	$\mathcal{F}(x, y) \downarrow$	$\mathcal{R}(.) \downarrow$	Realism ↑	Satisfaction ↑
SDEdit [23]	88.93	223.8	94.09 %	91.98%
Loopback [4]	104.6	132.9	54.28 %	85.32%
ILVR [5]	108.2	161.7	76.54 %	93.47%
Ours	94.40	134.2	N/A	N/A

Table 1. ***Quantitative Evaluations***. *(Left)* Method comparison w.r.t *faithfulness* $\mathcal{F}$ to the reference painting and *realism* $\mathcal{R}$ to the target domain. *(Right)* User-study results, showing % of inputs for which human subjects prefer our approach over prior works.

Baselines. We compare our approach with prior works on guided image synthesis from stroke paintings with no paired data. In particular we show comparisons with, *1) SDEdit* [23] wherein the generative prior is introduced by first passing the painting y through the forward diffusion pass $y \rightarrow y_{t_0}$ [16, 37], and then performing reverse diffusion $y_{t_0} \rightarrow y_0$ to get the output image $x = y_0$[1]. *2) SDEdit + Loopback* [4] which reuses the last diffusion output to iteratively increase the realism of the final output. *3) ILVR*[2] [23]: which uses an iterative refinement approach for conditioning the output x of the diffusion model with a guidance image y. Unless otherwise specified, we use the GradOP+ algorithm (refer Alg. 2) when reporting evaluation results.

Qualitative Results. Results are shown in Fig. 4. We observe that both proposed approximate optimization methods (*i.e. GradOP* in row-1,2 and *GradOP+* in row-3,4) lead to output images which are both highly photorealistic as well as *faithful* with reference painting. In contrast, while SDEdit [23] shows high faithfulness to the input painting, the final outputs lack details and resemble more of a pictorial art rather than realistic photos. Iteratively reperforming the guided synthesis with the generated outputs (SDEdit + Loopback [4]) helps improve the realism of output images, however, we find that this has two main disadvantages. First, the iterative loop increases the effective time required for generating each data sample (*e.g.* four reverse sampling steps instead of just one). Second, we note that as the number of successive iterations increase the final outputs become less and less faithful to the original painting input. Finally, ILVR [5] leads to more realistic outputs, however, the final outputs are not fully faithful to the reference painting in terms of the overall color composition.

[1] Please note that due to space constraints, we primarily use the standard hyperparameter value of $t_0 = 0.8$ in the main paper, and refer the reader to the supp. material for detailed comparisons with changing $t_0 \in [0, 1]$.

[2] Please note that the original ILVR [5] algorithm was proposed for iterative refinement with diffusion models in pixel space. We adapt the ILVR implementation for inference with latent diffusion models [30] for the purposes of this paper. Please refer supp. material for further details.

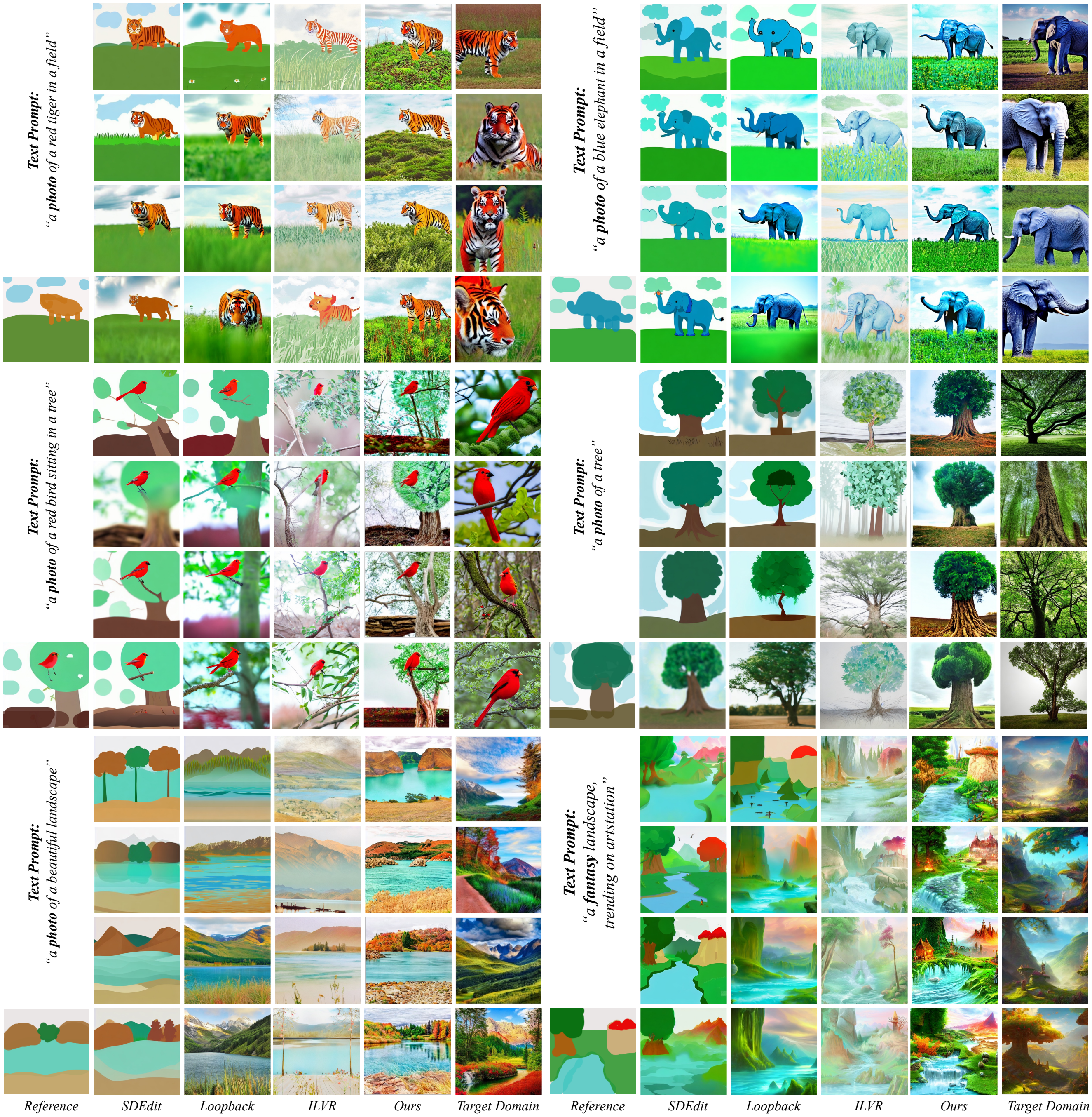

Text Prompt:
"a photo of a red tiger in a field"
Text Prompt:
"a photo of a blue elephant in a field"
Text Prompt:
"a photo of a red bird sitting in a tree"
Text Prompt:
"a photo of a tree"
Text Prompt:
"a photo of a beautiful landscape"
Text Prompt:
"a fantasy landscape, trending on artstation"
Reference
SDEdit
Loopback
ILVR
Ours
Target Domain
Reference
SDEdit
Loopback
ILVR
Ours
Target Domain

Figure 4. ***Qualitative comparisons.*** We compare the performance of our approach with prior works [4, 5, 23] based on their *faithfulness* to the provided reference, and the *realism* with respect to the target domain (generated by conditioning only on the text prompt). Please note that for our results, we show the *GradOP* (Alg. 1) and *GradOP+* (Alg. 2) outputs in row 1,2 and row-3,4 respectively.

Quantitative Results. In addition to qualitative results we also quantitatively evaluate the final outputs on the *faithfulness* $\mathcal{F}(x, y)$ and *targeted-realism* $\mathcal{R}(.)$ metrics defined earlier. Additionally, similar to [23] we also perform a human user study wherein the *realism* and the overall satisfaction score (*faithfulness* + *realism*) are evaluated by human subjects (please refer supp. material for details). Results are shown in Tab. 1. We find that as expected, while SDEdit [23] leads to the best faithfulness with the target painting, it exhibits very poor performance in terms of the *realism* score. SDEdit with loopback [4] improves the realism score but the resulting images start loosing faithfulness with the given reference. In contrast, our approach leads to the best tradeoff between faithfulness to the target im-

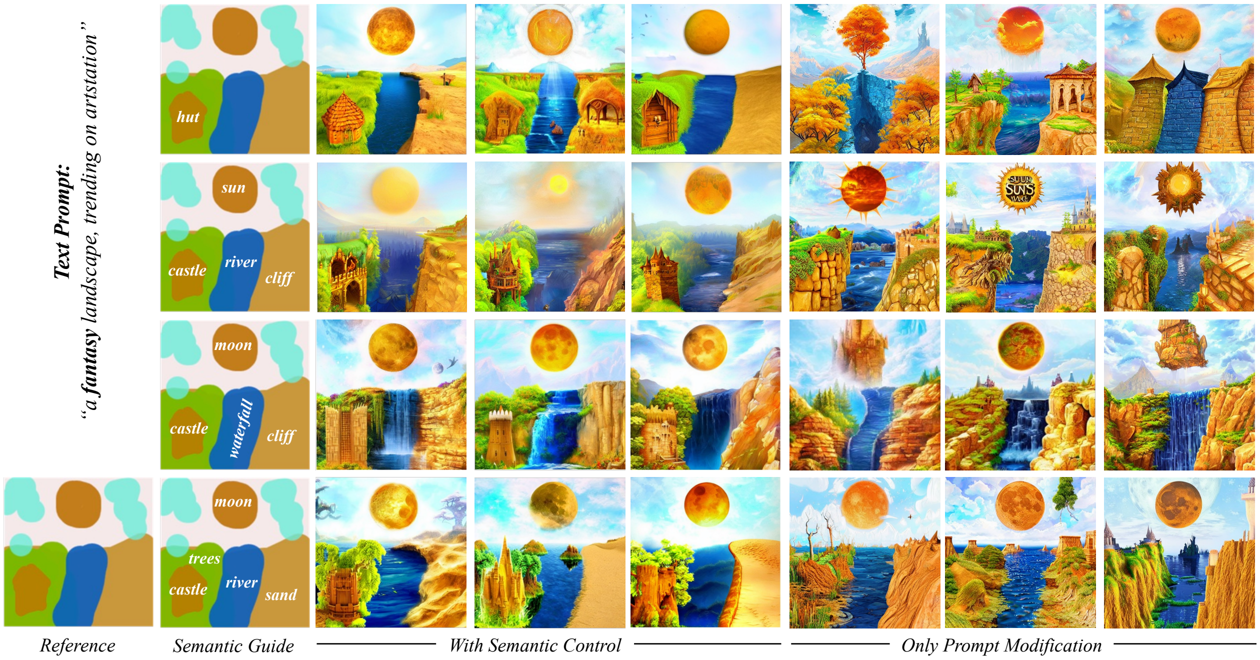


Figure 5. ***Controlling semantics of different painted regions.*** We compare image generation outputs (Col 2-4) using the cross-attention control approach from Sec. 3.3 with outputs (Col 5-7) generated by only modifying the input text prompt. Note that for each semantic guide (Col 2), the text prompt modification is performed by adding the corresponding semantic labels at the end of the text prompt. For instance, the modified text prompt for examples in row-1 would be "a fantasy landscape, trending on artstation showing a hut".

age and realism with respect to the target domain. These findings are also reflected in the user-study results wherein our method is preferred by $> 85.32\%$ of human subjects in terms of the overall satisfaction score.

4.2. Controlling Semantics of Painted Regions

Results are shown in Fig. 5. We observe that in absence of semantic attention control, the model tries to infer the semantics of different painting regions in an implicit manner. For instance, the orange strokes in the sky region can be inferred as the sun, moon, or even as a yellow tree. Similarly, the brown strokes in the lower-left region (intended to draw a *hut* or small *castle*) are often inferred as muddy or rocky parts of the terrain. Moreover, such disparity continues even after modifying the input prompt to describe the intended semantic labels. For instance, in row-1 from Fig. 5, while changing the text prompt to include the text "hut" leads to the emergence of "hut" like structures, the inference is often done in a manner that is not intended by the user.

In contrast, by ensuring a high overlap between the intended painting regions and the cross-attention maps for the corresponding semantic labels (refer Sec. 3.3), we are able to generate outputs which follow the intended semantic guide in a much more accurate manner. For instance, the user is able to explicitly specify that the brown regions on the ground describes a hut (row 1) or castle (row 2-4). Similarly, the semantics of different regions can be controlled, *e.g*. the blue region is specified as a river or waterfall, the orange strokes in the sky is specified as moon or sun *etc*.

5. Analysis

5.1. Variation in Target Domain

In this section, we analyse the generalizability of the our approach across different target domains (*e.g*. children drawings, disney scenes) and compare the output performance with prior works. Results are shown in Fig. 6-a. We observe that our approach is able to adapt the final image outputs reliably across a range of target domains while still maintaining a high level of faithfulness with the target image. In contrast, SDEdit [23] generates outputs which lack details and thereby look very similar across a range of target domains. SDEdit + Loopback [4] addresses this problem to some extent, but it requires multiple reverse diffusion passes and the generated outputs tend to lose their faithfulness to the provided reference with each iteration.

5.2. Variation with Number of Gradient Steps

In this section, we analyse the variation in output performance as we change the number of gradient descent steps N_{grad} used to solve the unconstrained optimization problem in Sec. 3. Results are shown in Fig. 6-b. As expected, we find that for $N_{grad} = 0$, the generated outputs are sampled randomly from the subspace of outputs $(\mathcal{S}_{\tau_{text}})$ conditioned only on the text. As the number of gradient-descent steps increase, the model converges to a subset of solutions

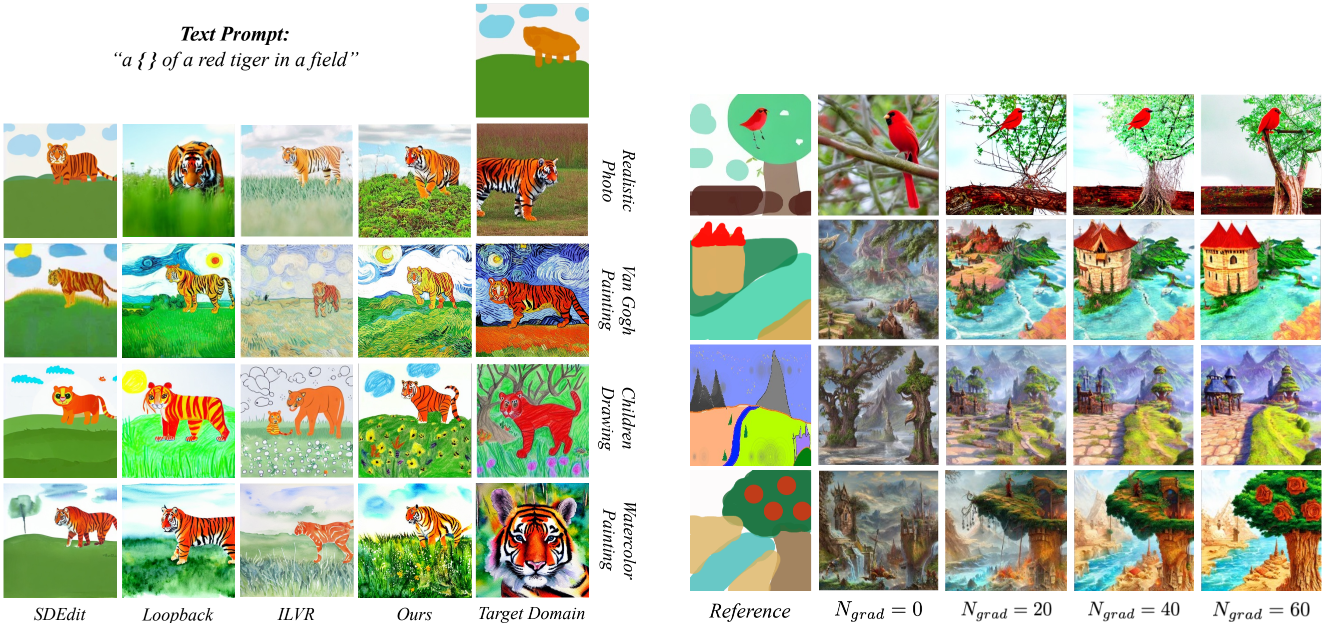


Figure 6. ***Method Analysis.*** Comparing guided image synthesis performance across *(left)* variation in target domain, and *(right)* variation in number of gradient descent steps N_{grad} used for performing the proposed optimization. Please zoom-in for best comparisons.

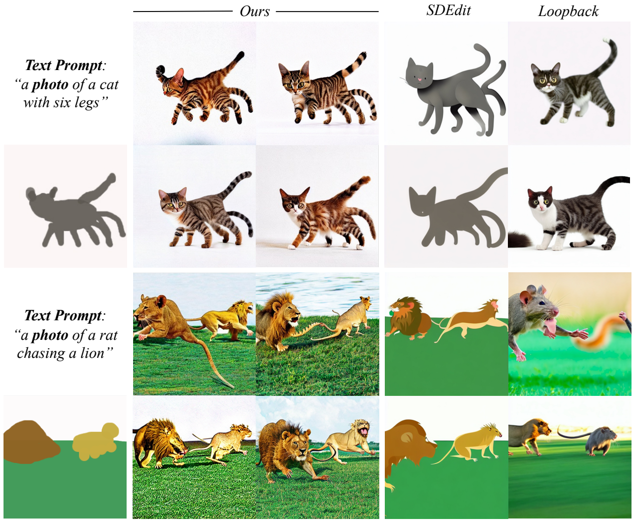

Ours
SDEdit
Loopback
Text Prompt:
“a photo of a cat with six legs”
Text Prompt:
“a photo of a rat chasing a lion”

Figure 7. ***Out-of-distribution performance.*** Analysing *success* (top) and *failure* (bottom) cases for out-of-distribution prompts.

within the target subsapce $\mathcal{S}_{\tau_{text}}$ which exhibit higher *faithfulness* with the provided reference. Please note that this behaviour is in contrast with SDEdit [23], wherein the increase in *faithfulness* to the reference is corresponded with a decrease in the *realism* of the generated outputs [23].

5.3. Out-of-Distribution Generalization

As shown in Sec. 4, 5, we find that the proposed approach allows for a high level of semantic control (both color composition and fine-grain semantics) over the output image attributes, while still maintaining the *realism* with respect to the target domain. Thus a natural question arises: *Can we use the proposed approach to generate realistic photos with out-of-distribution text prompts?*

As shown in Fig. 7, we observe that both success and failure cases exist for out-of-distribution prompts. For instance, while the model was able to generate “*realistic* photos of cats with six legs” (note that for the same inputs prior works either generate faithful but cartoon-like outputs, or, simply generate regular cats), it shows poor performance while generating “a photo of a rat chasing a lion”.

6. Conclusions

In this paper, we present a novel framework for performing guided image synthesis synthesis with user scribbles, without the need for paired annotation data. We point that prior works in this direction [4, 5, 23], typically adopt an inversion-like approach which leads to outputs which lack details and are often simplistic representations of the target domain. To address this, we propose a novel formulation which models the guided synthesis output as the solution of a constrained optimization problem. While obtaining an exact solution to this optimization is infeasible, we propose two methods *GradOP* and *GradOP+* which try to obtain an approximate solution to the constrained optimization in a sample-efficient manner. Additionally, we show that by defining a cross-attention based correspondence between the input text tokens and user painting, it is possible to control semantics of different painted regions without the need for semantic segmentation based conditional training.

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4432–4441, 2019. 2

[2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan++: How to edit the embedded images? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8296–8305, 2020. 2

[3] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Restyle: A residual-based stylegan encoder via iterative refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021. 2

[4] AUTOMATIC1111. Stable-diffusion-webui. `https://github.com/AUTOMATIC1111/stable-diffusion-webui`, 2022. 2, 5, 6, 7, 8, 15, 18

[5] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. *arXiv preprint arXiv:2108.02938*, 2021. 2, 5, 6, 8, 15, 16, 18

[6] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, et al. Cogview: Mastering text-to-image generation via transformers. *Advances in Neural Information Processing Systems*, 34:19822–19835, 2021. 2

[7] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12873–12883, 2021. 2

[8] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. *arXiv preprint arXiv:2203.13131*, 2022. 2

[9] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 2

[10] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 2, 11, 14, 17

[11] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 5, 16

[12] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 14, 15

[13] Nisha Huang, Fan Tang, Weiming Dong, and Changsheng Xu. Draw your art dream: Diverse digital art synthesis with multimodal guided diffusion. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 1085–1094, 2022. 2

[14] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 2

[15] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems*, 33:12104–12114, 2020. 2

[16] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2426–2435, 2022. 2, 3, 5, 11

[17] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5, 14

[18] Gihyun Kwon and Jong Chul Ye. Diffusion-based image translation using disentangled style and content representation. *arXiv preprint arXiv:2209.15264*, 2022. 2

[19] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5549–5558, 2020. 2

[20] Songhua Liu, Tianwei Lin, Dongliang He, Fu Li, Ruifeng Deng, Xin Li, Errui Ding, and Hao Wang. Paint transformer: Feed forward neural painting with stroke prediction. *arXiv preprint arXiv:2108.03798*, 2021. 2

[21] Xihui Liu, Dong Huk Park, Samaneh Azadi, Gong Zhang, Arman Chopikyan, Yuxiao Hu, Humphrey Shi, Anna Rohrbach, and Trevor Darrell. More control for free! image synthesis with semantic diffusion guidance. *arXiv preprint arXiv:2112.05744*, 2021. 2

[22] Xihui Liu, Guojun Yin, Jing Shao, Xiaogang Wang, et al. Learning to predict layout-to-image conditional convolutions for semantic image synthesis. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[23] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. 2, 5, 6, 7, 8, 11, 12, 13, 14, 15, 16, 17, 18, 19

[24] Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. Mixed precision training. *arXiv preprint arXiv:1710.03740*, 2017. 18

[25] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 1, 2

[26] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2337–2346, 2019. 2

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry,

Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 5

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

[29] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021. 2

[30] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 1, 2, 5, 14, 15

[31] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022. 2

[32] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 1, 2

[33] Junyoung Seo, Gyuseong Lee, Seokju Cho, Jiyoung Lee, and Seungryong Kim. Midms: Matching interleaved diffusion models for exemplar-based image translation. *arXiv preprint arXiv:2209.11047*, 2022. 2

[34] Jaskirat Singh, Cameron Smith, Jose Echevarria, and Liang Zheng. Intelli-paint: Towards developing human-like painting agents. In *European conference on computer vision*. Springer, 2022. 2

[35] Jaskirat Singh and Liang Zheng. Combining semantic guidance and deep reinforcement learning for generating human level paintings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 2

[36] Jaskirat Singh, Liang Zheng, Cameron Smith, and Jose Echevarria. Paint2pix: Interactive painting based progressive image synthesis and editing. In *European conference on computer vision*. Springer, 2022. 2

[37] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2, 3, 5, 11, 15, 18

[38] Vadim Sushko, Edgar Schönfeld, Dan Zhang, Juergen Gall, Bernt Schiele, and Anna Khoreva. You only need adversarial supervision for semantic image synthesis. *arXiv preprint arXiv:2012.04781*, 2020. 2

[39] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *arXiv preprint arXiv:2102.02766*, 2021. 2

[40] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers, 2022. 5, 14, 15

[41] Weilun Wang, Jianmin Bao, Wengang Zhou, Dongdong Chen, Dong Chen, Lu Yuan, and Houqiang Li. Semantic image synthesis via diffusion models. *arXiv preprint arXiv:2207.00050*, 2022. 2

[42] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022. 1, 2

[43] Yufan Zhou, Ruiyi Zhang, Changyou Chen, Chunyuan Li, Chris Tensmeyer, Tong Yu, Jiuxiang Gu, Jinhui Xu, and Tong Sun. Towards language-free training for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17907–17917, 2022. 2

[44] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A Efros. Generative visual manipulation on the natural image manifold. In *European conference on computer vision*, pages 597–613. Springer, 2016. 2

[45] Peihao Zhu, Rameen Abdal, Yipeng Qin, and Peter Wonka. Sean: Image synthesis with semantic region-adaptive normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5104–5113, 2020. 2

[46] Zhengxia Zou, Tianyang Shi, Shuang Qiu, Yi Yuan, and Zhenwei Shi. Stylized neural painting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15689–15698, 2021. 2

Supplementary Material
High-Fidelity Guided Image Synthesis with Latent Diffusion Models

A. Additional Results

In this section, we provide additional results which could not be included in the main paper due to space constraints. In particular, we note that baseline methods like SDEdit [23] can often be run using different values of the hyperparameter t_0. We therefore provide additional results comparing the performance of SDEdit at different $t_0 \in [0, 1]$ (refer Sec. A.1). Additionally, we introduce some custom baselines (which could be used for improving the realism of final image outputs) and show results comparing their output performance with our approach (refer Sec. A.2).

A.1. Additional Comparisons with SDEdit

Recall, given a stroke painting y, SDEdit [23] follows an inversion-based approach for performing guided image synthesis. In particular, the generative prior is introduced by first passing the painting y through the forward diffusion pass $y \rightarrow y_{t_0}$ [16,37], and then performing reverse diffusion $y_{t_0} \rightarrow y_0$ to get the output image $x = y_0$. Due to space constraints, we primarily use the standard hyperparameter value of $t_0 = 0.8$ in the main paper. In this section, we provide additional results which comprehensively compare our approach with SDEdit [23] under changing values of t_0.

Qualitative Comparisons. Results are shown in Fig. 9, 10. We observe that for lower values of t_0, SDEdit generates outputs which though highly faithful to the reference painting, lack details and represent simplistic representations of the target domain. Increasing the value of hyperparameter t_0 helps improve realism but the outputs become less and less faithful with the reference painting. In contrast, we find that the proposed approach leads to outputs which are both *faithful* to the reference painting as well as exhibit high *realism w.r.t* the target domain (which is generated using only text prompt conditioning).

Quantitative Comparisons. In addition to qualitative results, we also report quantitative results by analysing the relationship between the *faithfulness* $\mathcal{F}$ and *realism* $\mathcal{R}$ metrics (refer Sec. 4.1 of main paper), under changing values hyperparameter t_0. Results are shown in Fig. 8. We observe that as compared to prior works, our method provides the best tradeoff generating *realistic* outputs and maintaining *faithfulness* with the provided reference painting.

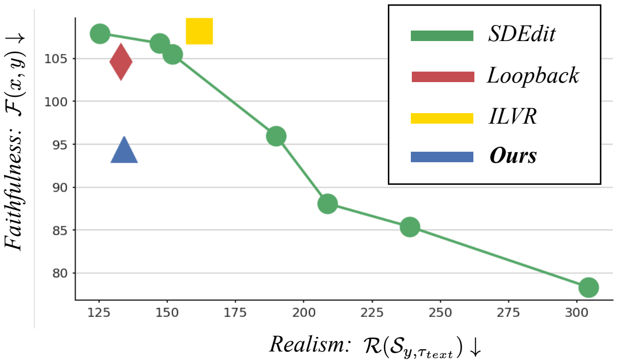


Figure 8. ***Visualizing faithfulness-realism tradeoff.*** We analyse the tradeoff between faithfulness-realism distances for different methods (note that lower is better for both metrics). We observe that as compared to prior works, our method provides the best tradeoff between generating realistic outputs and maintaining faithfulness with the provided reference painting.

A.2. Comparison with Custom Baselines

In this section, we introduce some custom methods (as baselines) for increasing the realism of generated outputs with SDEdit [23], and then compare the output performance for the same with our approach. In particular, we show additional comparisons with the following custom baselines,

- ***Attention Re-weighting (AttnRW)*** [10] wherein the realism *w.r.t* the target domain is enhanced by increasing the attention weighting for the corresponding domain specific text tokens (*e.g.* photo, painting *etc*.). For instance, if the text prompt says *"a photo of a tree"*, then we aim to increase the realism of the generated outputs by increasing the weightage of the cross-attention maps corresponding to the the word *"photo"* [10]. Results are shown in Fig. 11. We observe that while increasing the weightage of domain specific text tokens (*e.g.* photo, painting *etc*.) helps improve the realism of the output images to some extent, the final images still lack details and certain blurry regions still persist (*e.g.* grass in row-1). Furthermore, the increase in realism is accompanied by some image artifacts *e.g.* blotched image regions in row 1-4, image-in-image artifacts in row 4-8 *etc*. In contrast, we find that our method provides a more practical approach for increasing

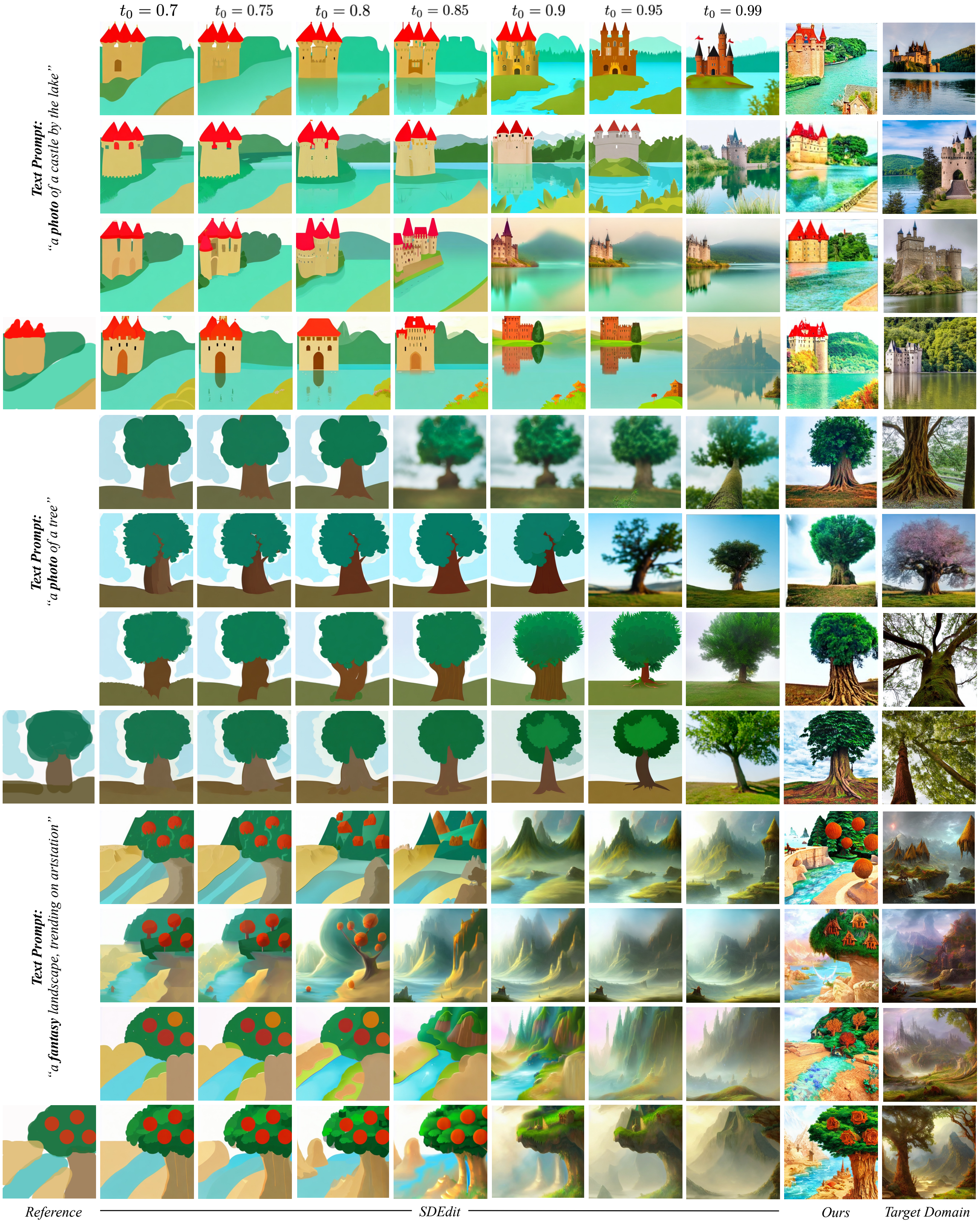


Figure 9. ***Additional comparisons.*** We provide comprehensive comparisons with SDEdit [23] under changing value of hyperparameter t_0. We find that SDEdit [23] either generates faithful but cartoon-like outputs for low t_0, or, generates realistic but unfaithful outputs at high t_0. In contrast, our approach leads to outputs which are both realistic (*w.r.t* the target domain) as well as faithful (to the provided reference).

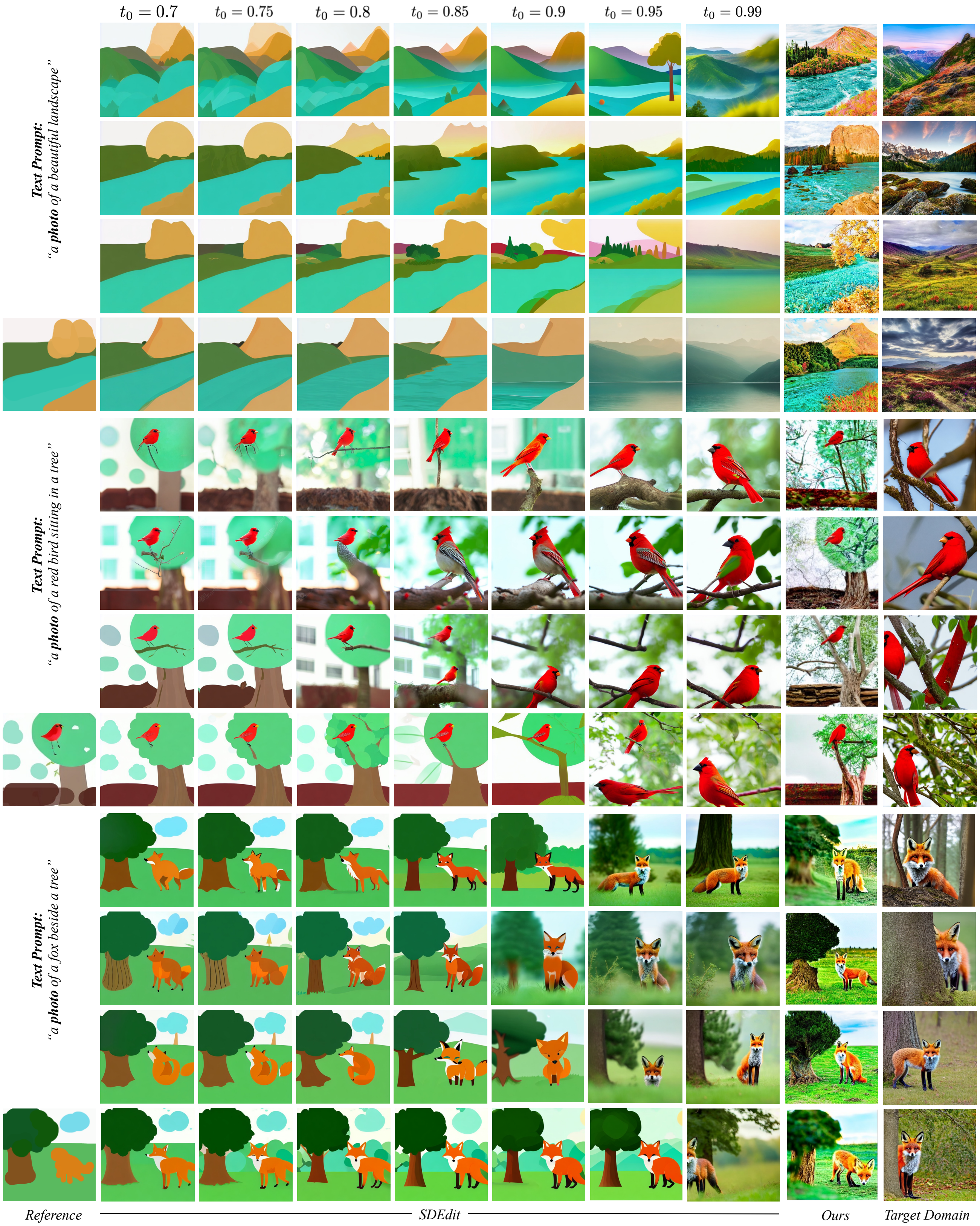

$t_0 = 0.7$
$t_0 = 0.75$
$t_0 = 0.8$
$t_0 = 0.85$
$t_0 = 0.9$
$t_0 = 0.95$
$t_0 = 0.99$
Text Prompt: "a photo of a beautiful landscape"
Text Prompt: "a photo of a red bird sitting in a tree"
Text Prompt: "a photo of a fox beside a tree"
Reference
SDEdit
Ours
Target Domain

Figure 10. ***Additional comparisons.*** We provide comprehensive comparisons with SDEdit [23] under changing value of hyperparameter t_0. We find that SDEdit [23] either generates faithful but cartoon-like outputs for low t_0, or, generates realistic but unfaithful outputs at high t_0. In contrast, our approach leads to outputs which are both realistic (*w.r.t* the target domain) as well as faithful (to provided reference).

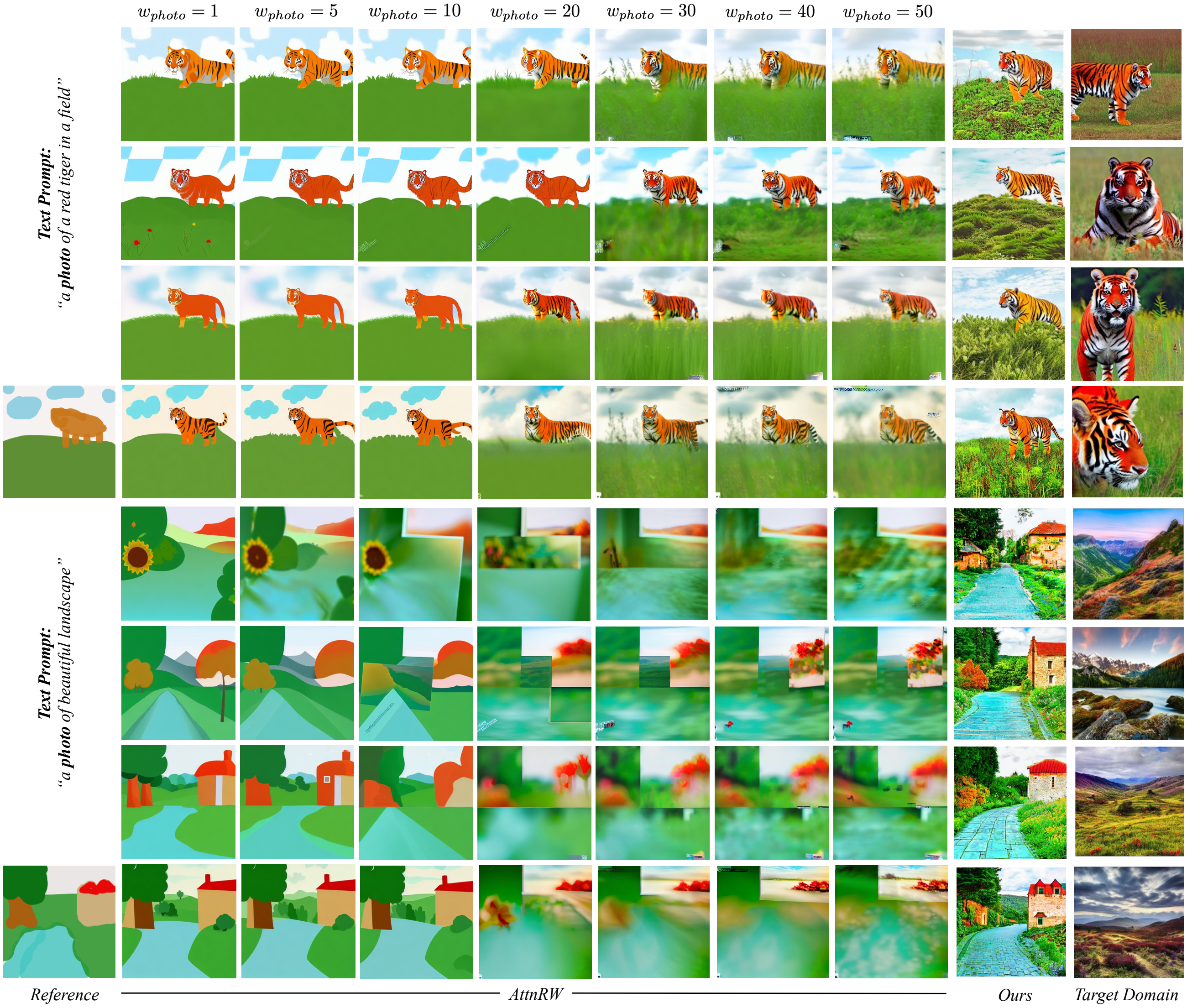


Figure 11. ***Comparison with Custom Baselines - AttnRW*** *[10].* We compare the performance of our method with the Attention Reweighting (AttnRW) approach for increasing realism *w.r.t* the target domain. We find that increasing the weight of cross attention maps corresponding to the domain-specific text tokens (*e.g.* photo in above), leads to improved realism of the generated outputs. However, we note that certain blurry details persist *e.g.* grass in row 1-4. Also, the increase in realism is accompanied by some image artifacts *e.g.* blotched regions in row 1-4, image in image artifacts in row 4-8 *etc*. In contrast, our approach improves output realism in a more coherent manner.

the output realism in a semantically coherent manner.

- ***Increasing Classifier Guidance Scale*** [12], wherein we attempt to increase the realism of the SDEdit [23] outputs by increasing the scale of classifier free guidance used during the reverse diffusion process. Results are shown in Fig. 12. We observe that while increasing the scale of classifier free guidance improves the level of detail in the generated images, the final outputs still resemble cartoon-like or simplistic representations of the target domain. Furthermore, we also note that our approach can also benefit from the increase in guidance scale in order to increase the level of fine-grain detail in the output images (*e.g.* details of castle, water reflections in Fig. 12).

B. Experiment Details

B.1. Implementation Details

In this section, we provide further details for the implementation of our approach as well as other baselines used while reporting results in the main paper.

Ours. We use publicly available text-conditioned latent diffusion models [30,40] for implementing the purposed approach in the main paper. The constrained optimization is performed using gradient descent with the Adam [17] optimizer and number of gradient steps $N_{grad} \in [20, 60]$. While several formulations of the distance measure $\mathcal{L}$ and painting function f are possible (refer Sec. C), we find that

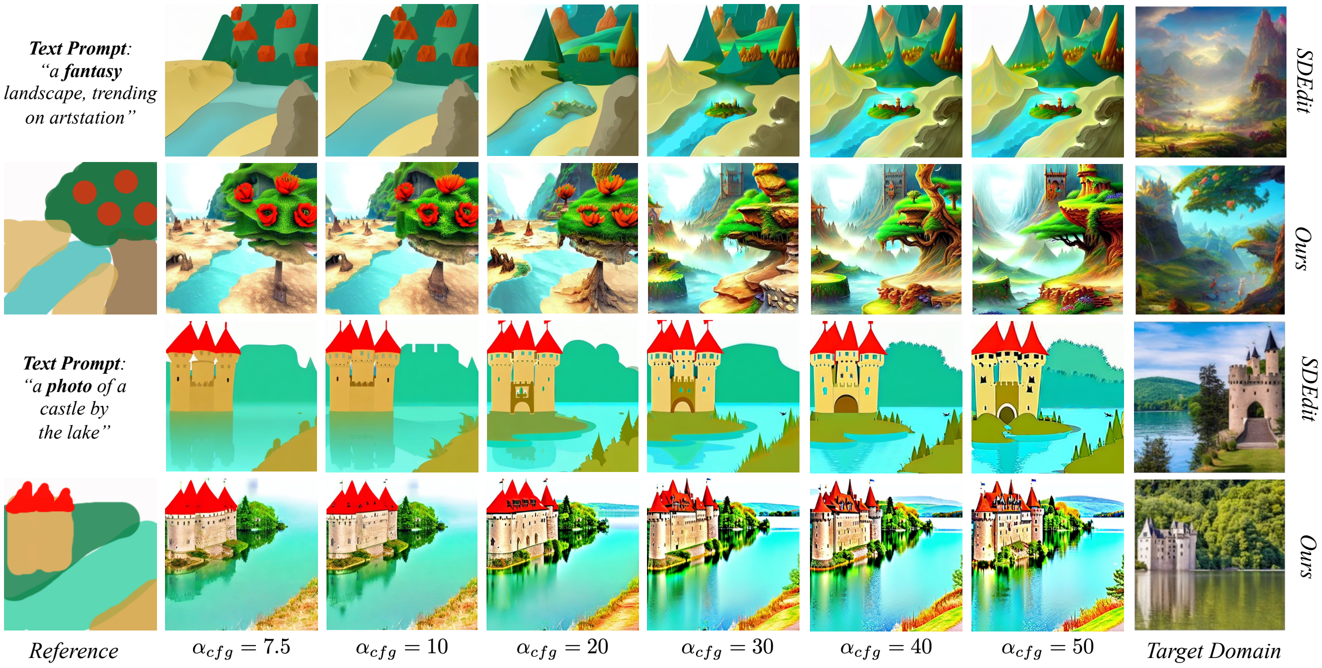


Figure 12. ***Comparison with Custom Baselines - CFG*** [12]. We analyse the impact of increasing the classifier-free guidance scale α_{cfg} on outputs generated using SDEdit [23] and our method. We find that while increasing the value of α_{cfg} leads to increase in level of details, the final outputs still represent simplistic representations of the target domain (row-3). Furthermore, as the value of α_{cfg} is increased, the faithfulness with respect to the reference painting is compromised (*e.g.* red regions in row-1).

simply approximating the function $\mathcal{L}$ using mean squared distance and f as a convolution operation with a gaussian kernel seems to give the fastest inference time performance with our method. For consistency with prior works, we use the non-differentiable painting function from SDEdit [23] while reporting quantitative results. All results are reported using the DDIM sampling [37] with 50 inference steps for performing the reverse diffusion process.

SDEdit [23]. We use the standard image-to-image pipeline from the open-source *diffusers* library [40] for reporting results for *SDEdit* [23] with different values of hyperparameter $t_0 \in [0, 1]$. Similar to our method, all results are reported at 512×512 resolution using DDIM sampling [37] with 50 inference steps for performing the reverse diffusion process. Unless otherwise specified, a classifier-free guidance scale [12] of 7.5 is used for all experiments.

SDEdit + Loopback [4]. We use the previously described SDEdit implementation and iteratively reperform guided synthesis with the previous diffusion outputs to improve realism of the generated outputs. In particular, we use $N_{iter} = 4$ iterations for the iterative process. Also, similar to [4], in order to increase the realism of generated outputs with each iteration, the hyperparameter t_0 is updated as,

$$t_0^{n+1} \leftarrow min(t_0^n \cdot k, 1.0), \quad k \in [1.0, 1.1] \tag{11}$$

where $n \in [1, N_{iter}]$ is the iteration number. Unless otherwise specified, we use the standard hyperparameter selection of $k = 1.05$ and $t_0^{n=1} = 0.8$ for our experiments.

ILVR [5]. The original ILVR [5] algorithm was proposed for iterative refinement with diffusion models in pixel space. We adapt the ILVR implementation for inference with latent diffusion models [30] for the purposes of this paper. In particular given a reference painting y, the original ILVR algorithm modifies the diffusion output x_t (in pixel space) at any timestep t during reverse diffusion process as,

$$\tilde{x}_t = \phi_N(y_t) + x_t - \phi_N(x_t), \quad y_t \sim q(y_t \mid y) \tag{12}$$

where $q(y_t \mid y)$ represents the forward diffusion process from $y \rightarrow y_t$, $\phi_N(.)$ is a low pass filter achieved by scaling down the image by a factor of N and then upsampling it back to the original dimensions. Assuming a latent diffusion model with encoder $\mathcal{E}$ and decoder $\mathcal{D}$, we simply adapt the above update in latent space as follows,

$$x_t = \mathcal{D}(z_t) \tag{13}$$

$$z_y = \mathcal{E}(y), \ z_{y_t} \sim q(z_{y_t} \mid z_y) \tag{14}$$

$$\tilde{x}_t = \phi_N(y_t) + x_t - \phi_N(x_t), \quad y_t = \mathcal{D}(z_{y_t}) \tag{15}$$

$$\tilde{z}_t = \mathcal{E}(\tilde{x}_t) \tag{16}$$

where Eq. 13, 16 map the latent features z_t to pixel space x_t, and vice-versa. Eq. 14 computes y_t from y by first mapping y to z_y, computing the forward diffusion $z_y \rightarrow z_{y_t}$ and then reverting back z_{y_t} to y_t. Finally, Eq. 15 is simply the original update rule from ILVR algorithm [5]. A hyperparmeter value of $N = 4$ is used while reporting results.

B.2. Quantitative Experiments

Data Collection. Since there is no predefined dataset for guided image synthesis with user-scribbles and text prompts, we create our own dataset for reporting quantitative results. In particular, we first collect a set of 100 stroke painting and text prompt pairs from diverse data modalities with the help of actual human users. We then augment the collected data using a prompt-engineering approach to increase the diversity of the collected data pairs. In particular, the text prompt for each data-pair is modified in order to in order to replace the domain specific text words (*e.g.* photo, painting) with pre-designed target domain templates, while keeping the underlying content the same. During prompt engineering, the target domain template is chosen randomly from *['photo','watercolor painting', 'Vincent Van Gogh painting','children drawing','high resolution disney scene', 'high resolution anime scene', 'fantasy scene','colored pencil sketch']*. For each data pair, we then sample four random guided image synthesis outputs for each baseline and our method. The resulting dataset consists of 800 (painting, text-prompt) pairs and 3200 overall samples from diverse data modalities for final method evaluation.

Quantitative Metrics. In order to evaluate the performance of our approach, we introduce two metrics for measuring the *faithfulness* of the output *w.r.t* the reference painting, and the *realism* of the generated samples *w.r.t* the target domain (specified through text-only conditioning). In particular, given an input painting y and output real image prediction x, we define faithfulness distance $\mathcal{F}(x, y)$ as,

$$\mathcal{F}(x, y) = \mathcal{L}_2(f(x), y) \tag{17}$$

where $f(.)$ is the painting function. Thus an output image x is said to have high faithfulness with the given painting y if upon painting the final output x we get a painting $\tilde{y} = f(x)$ which is similar to the original target painting y (Fig. 13).

The painting function f is implemented using the human stroke-simulation algorithm from SDEdit [23]. In particular, given an 256×256 input image, the output painting is computed by first passing the image through a median filter with kernel size 23, and then perform color quantization to reduce the number of colors to 20 using an adaptive palette.

Similarly, given a set of output data samples $\mathcal{S}(y, \tau_{text})$ conditioned on both painting y and text τ_{text}, and, $\mathcal{S}(\tau_{text})$ conditioned only on the text, the *realism* $\mathcal{R}$ is defined as,

$$\mathcal{R}(\mathcal{S}(y, \tau_{text})) = FID\left(\mathcal{S}(y, \tau_{text}), \mathcal{S}(\tau_{text})\right) \tag{18}$$

where FID represents the Fisher inception distance [11].

Please note that while the above defined *realism* distance measure $\mathcal{R}$ captures the realism with respect to the target domain, we expect the computed FID scores to be higher than those expected of unconditioned image outputs. This is because while the proposed method generates outputs which seem realistic to human eyes, the variance of output distribution is significantly lower than that of real images. The decreased variance in output images occurs simply because the layout and color composition are predominantly fixed as a result of additional conditioning on the stroke painting y. In contrast, natural images or images conditioned only on the text prompt have a much higher diversity in terms of scene layout and the overall color composition. We therefore try to overcome of lack of diversity in generated image outputs by performing random data augmentations (random horizontal flip and random resized crop of size 448×448 on a 512×512 image) while computing the final realism scores across different methods [3].

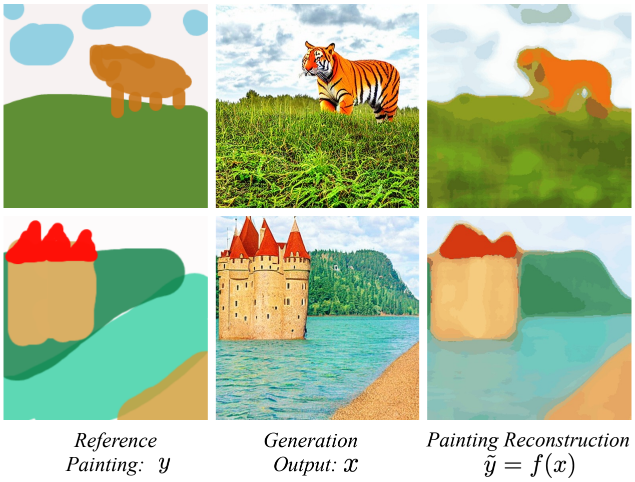


Figure 13. Visualizing input painting y, output x and painted reconstruction $\tilde{y} = f(x)$. The goal is to generate an output x which is realistic and for which painting loss $\mathcal{L}_2(f(x), y)$ is minimized.

Human User Study. In addition to reporting quantitative results using the above defined measures for *faithfulness* and *realism*, similar to [23], we also perform a human user study wherein the *realism* and the overall satisfaction score (*faithfulness* + *realism*) are evaluated by actual human users. For the *realism* scores, given an input text prompt (with target domain τ_{domain} *e.g.* τ_{domain} = '*photo*') and sample images conditioned only on the text prompt, the participants were shown a pair of image generation outputs comparing our method with prior works. For each pair, the human subject is then asked to select the output image which is more realistic with respect to the target domain (τ_{domain}). Similarly, for computing the overall satisfaction scores, given an input stroke painting, text prompt and sample images conditioned only on the text

[3] Note that while this helps increase the diversity in scene layout the diversity in color composition is still lower than that of real images or image outputs conditioned only on the text prompt.

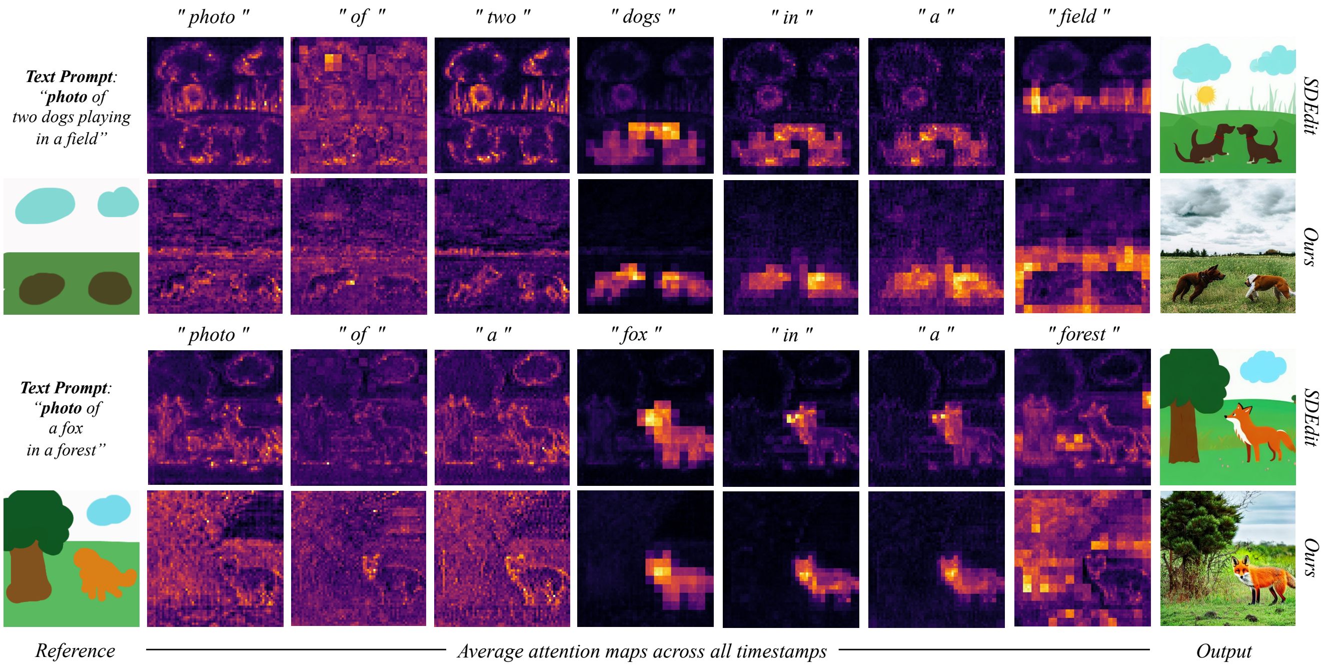


Figure 14. ***Visualizing the effect of GradOP on cross attention maps.*** We analyse the effect of our approach on the cross-attention maps generated during the reverse diffusion process. We find that our method leads to cross-attention outputs which help the model pay better attention to desired image areas in the reference painting. For instance, in the first example, the cross-attention features show high overlap with the desired *dog* and *field* regions. In contrast, the cross attention maps from SDEdit [23] reveal that the model is not paying adequate attention to the desired image areas (*e.g. field* in row-1, *tree and forest* in row-3) while generating the final output.

prompt, the participants were shown a pair of image generation outputs comparing our method with prior works. For each pair, the instruction is: "Given the input painting and text prompt, how would you imagine this image to look like in reality? Your selection should be based on how realistic and less blurry the image is (please check level of details), consistency with the target domain (τ_{domain}) and whether it is faithful with the reference painting in terms of scene layout, color composition". For each task (*e.g.* computing overall satisfaction score), the collected data samples (discussed above) were divided among 50 human participants, who were given an unlimited time in order to ensure high quality of the final results. Additionally, in order to remove data noise, we use a repeated comparison (control seed) for each user. Responses of users who answer differently to this repeated seed are discarded while reporting the final results.

C. Method Analysis: Continued

C.1. Effect of GradOP on Cross Attention Maps

As shown by Hertz *et al.* [10] and our results, the cross-attention maps corresponding to different words in the input text prompt play a key role in deciding the overall semantic contents of the final image output. In this section, we try to analyse how the proposed approach leads to more realistic image content generation by analysing the average cross-attention maps generated while performing the reverse diffusion process with SDEdit [23] and our method.

Results are shown in Fig. 14. We find that our method leads to cross-attention outputs which help the model pay better attention to desired image areas in the reference painting. For instance, in the first example, the cross-attention features show high overlap with the desired *dog* and *field* regions. In contrast, the cross attention maps from SDEdit [23] reveal that the model is not paying adequate attention to the some desired image areas (*e.g. field* in row-1, *forest* in row-3) while generating the final output.

C.2. Semantic Control without Painting Guidance

Recall that in addition to performing *high-fidelity* guided image synthesis, we also show that by simply defining a cross attention based correspondence between the input text tokens and the user painting, it is possible to control the semantics of different image regions without the need for any semantic segmentation based conditional training. In this section, we analyse whether similar semantic control is possible without having additional guidance through a stroke painting. In particular, we wish to analyse if such fine-grain control is only possible while providing additional guidance through the reference stroke painting?

To answer this question, we compare the outputs generated through semantic control with and without using a

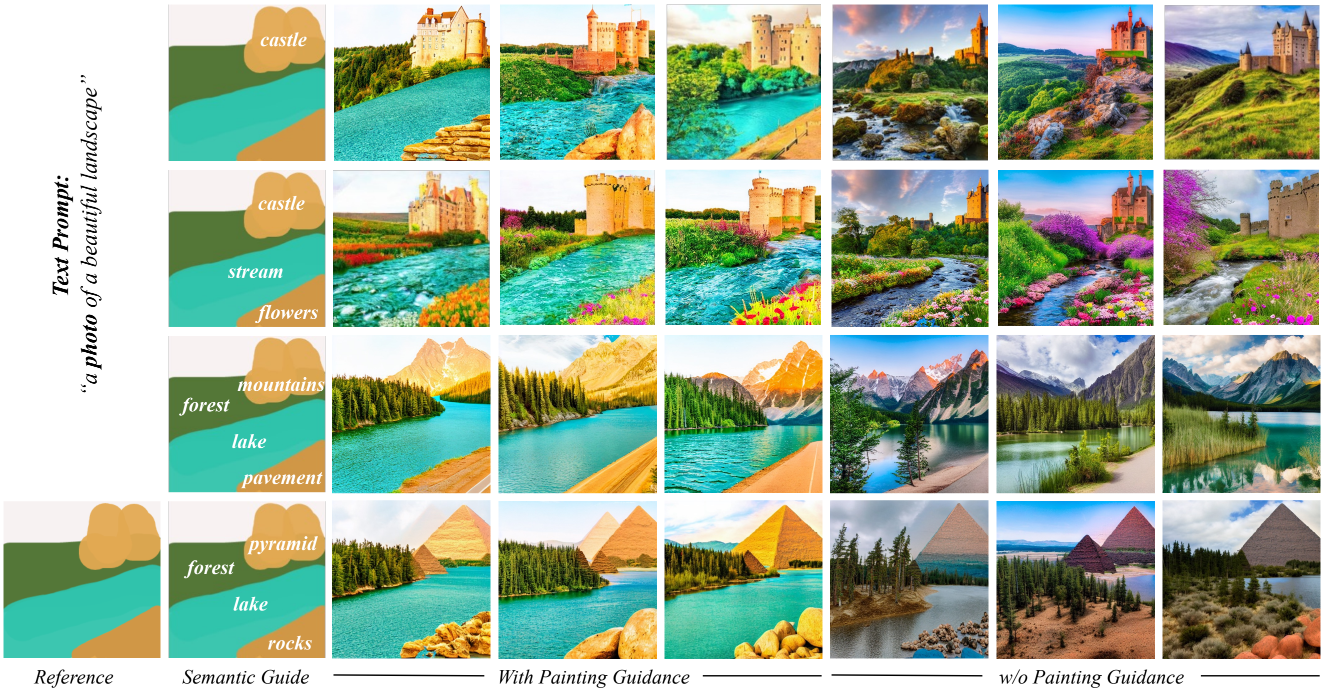


Figure 15. ***Analysing role of painting guidance in semantic control.*** We analyse the effect of using an underlying reference painting as guidance in controlling the semantics of different image areas using cross-attention based correspondence approach presented in the main paper (refer Sec. 3.3 in main paper). We find that additional guidance using reference stroke painting helps the user gain much accurate control over the semantics of different image regions (*e.g*. lake in row-3,4, mountains in row-3, rocks, forest in row-4 *etc*.).

reference painting for the guided synthesis process. Results are shown in Fig. 15. We observe that while for it is feasible to define the semantics of one or two parts of the image accurately using *cross-attention* correspondence, the performance decreases as the number of semantic labels increases (*e.g*. lake in row-3,4, mountains in row-3, rocks, forest in row-4 *etc*.). In contrast, we find that the use of a reference painting results in much better control over the semantics of different image regions. We believe that the same is because the use of a reference painting sets up a generic semantic structure for the output image which can then be easily refined by defining a cross-attention based correspondence. For instance, in row-4 of Fig. 15, adding the blue strokes for lake region sets up a semantic prior which constrains the inference of output semantics to semantic categories like river, lake, sea, stream, blue-green grass, blue pavement *etc*. The use of semantic correspondence then helps refine these output semantics to what is actually desired by the user. In contrast, without stroke guidance, the initial semantics for lake region could me much more diverse (*e.g*. sand, rocky terrain in row-4), and thereby more challenging to refine through the proposed semantic correspondence strategy.

C.3. Inference Time Analysis

We report a comparison of the average inference times required for each output image in Tab. 2. All results are reported using the DDIM sampling [37] with 50 inference steps, on a single Nvidia RTX 3090 GPU.

Method	Inference Time (s)	
	w/o mixed precision	*with mixed precision* [24]
SDEdit [23]	6.32 s	4.45 s
Loopback [4]	27.2 s	20.46 s
ILVR [5]	8.24 s	6.17 s
GradOP **(Ours)**	20.1 s	15.8 s
GradOP+ **(Ours)**	12.3 s	8.86 s

Table 2. ***Inference time analysis***. Comparing inference time required for generating each output image for different methods. All results are reported with DDIM sampling and 50 inference steps.

C.4. Variation in Painting Function

Please recall that a key requirement for solving the proposed constrained optimization in Sec. 3 is to define a differentiable painting function f, which provides a good approximation for *"how a human would paint a given image with coarse user-scribbles"*. In this section, we therefore look at some possible formulations for obtaining an approximation of the painting function in a differentiable manner, and compare the corresponding output results.

Painting Function Formulation. In particular, we consider three main formulations for constructing a differentiable painting function f, 1) *Median Filter + Color Quan-*

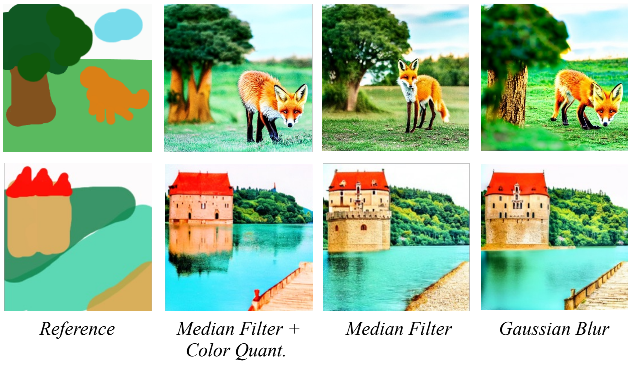


Figure 16. Analysing performance for different differentiable approximations of the painting function f. We find that while using a more accurate painting function [23] (Col-2) leads to slightly more details (*e.g*. notice the gradient of the grass regions in row-1, detailed shadows of the castle and island in row-2), in practice more simpler approximations (*e.g*. Gaussian Blur) also produces highly realistic outputs while allowing for much faster inference times.

tization, wherein we implement a differentiable approximation of the human-stroke simulation algorithm in [23]. In particular, given a reference painting y and output x, we first pass x through a median filter of size 23. We then pass the output of the last step through a differentiable color quantization function which maps the image pixels to their nearest rgb value in the painting y (that is, we are performing color quantization *w.r.t* the palette of the reference painting.) 2) *Median Filter* wherein we use the median filter alone for approximating the painting function, and 3) *Gaussian Blur* wherein approximate the painting function through a convolution operation with a Gaussian kernel (size 31 and σ=7).

Results are shown in Fig. 16. We observe that while the use of a more accurate human-stroke simulation function from [23] allows for the generation of slightly more detailed outputs (*e.g*. notice the gradient of the grass regions in row-1, detailed shadows of the castle and island in row-2), it increases the overall inference time required for the proposed gradient descent optimization (40.7s on *GradOP+*). In contrast, we find that using much more simpler approximations (*e.g*. Median Filter, Gaussian Blur) for the painting function also produces highly realistic outputs while allowing for much faster inference times (8.86s, 14.1s on *GradOP+* for Gaussian Blur and Median Filter respectively).