SmartBrush: Text and Shape Guided Object Inpainting with Diffusion Model

Shaoan Xie [1]*, Zhifei Zhang[2], Zhe Lin[2], Tobias Hinz[2], Kun Zhang[1,3]
[1]Carnegie Mellon University
[2]Adobe Research
[3]Mohamed bin Zayed University of Artificial Intelligence
shaoan@cmu.edu, {zzhang, zlin, thinz}@adobe.com, kunz1@cmu.edu

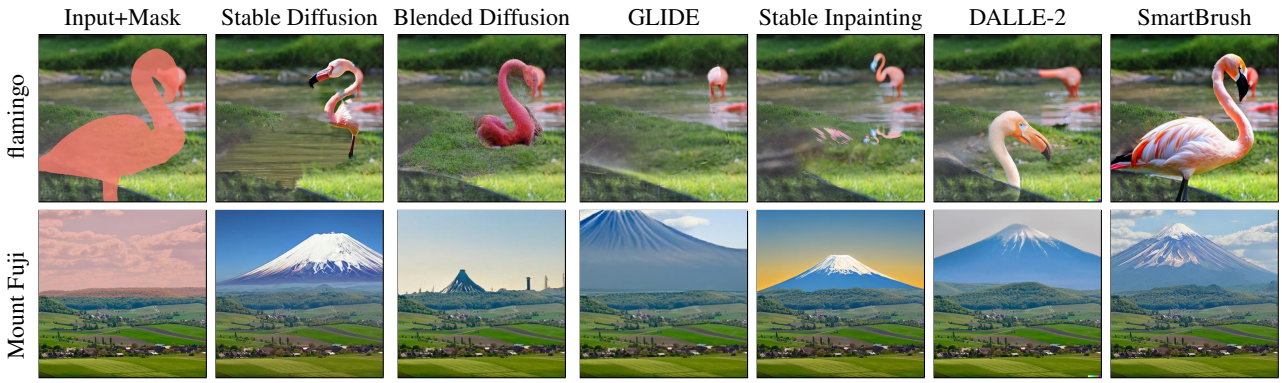

Figure 1. Our method generates high-quality object inpainting results. Different mask precision levels allowing users to either provide exact masks (top row) or to use a rough mask outline (bottom row). Compared to existing methods, our method generates more realistic images, follows accurate masks more closely (top row) and shows better background preservation for coarse masks (bottom row).

Abstract

Generic image inpainting aims to complete a corrupted image by borrowing surrounding information, which barely generates novel content. By contrast, multi-modal inpainting provides more flexible and useful controls on the inpainted content, e.g.*, a text prompt can be used to describe an object with richer attributes, and a mask can be used to constrain the shape of the inpainted object rather than being only considered as a missing area. We propose a new diffusion-based model named SmartBrush for completing a missing region with an object using both text and shape-guidance. While previous work such as DALLE-2 and Stable Diffusion can do text-guided inapinting they do not support shape guidance and tend to modify background texture surrounding the generated object. Our model incorporates both text and shape guidance with precision control. To preserve the background better, we propose a novel training and sampling strategy by augmenting the diffusion U-net with object-mask prediction. Lastly, we introduce a multi-task training strategy by jointly training inpainting with text-to-image generation to leverage more training data. We conduct extensive experiments showing that our model outperforms all baselines in terms of visual quality, mask controllability, and background preservation.*

1. Introduction

Traditional image inpainting aims to fill the missing area in images conditioned on surrounding pixels, lacking control over the inpainted content. To alleviate this, multi-modal image inpainting offers more control through additional information, *e.g.* class labels, text descriptions, segmentation maps, *etc*. In this paper, we consider the task of multi-modal object inpainting conditioned on both a text description and the shape of the object to be inpainted (see Fig. 1). In particular, we explore diffusion models for this task inspired by their superior performance in modeling complex image distributions and generating high-quality images.

Diffusion models (DMs) [7, 24], *e.g.*, Stable Diffusion [20], DALL-E [18, 19], and Imagen [21] have shown promising results in text-to-image generation. They can

* Work done during internship at Adobe.

also be adapted to the inpainting task by replacing the random noise in the background region with a noisy version of the original image during the diffusion reverse process [14]. However, this leads to undesirable samples since the model cannot see the global context during sampling [16]. To address this, GLIDE [16] and Stable Inpainting (inpainting specialist v1.5 from Stable Diffusion) [20] randomly erase part of the image and fine-tune the model to recover the missing area conditioned on the corresponding image caption. However, semantic misalignment between the missing area (local content) and global text description may cause the model to fill in the masked region with background instead of precisely following the text prompt as shown in Fig. 1 ("Glide" and "Stable Inpainting"). We refer to this phenomenon as ***text misalignment***.

An alternative way to perform multi-modal image inpainting is to utilize powerful language-vision models, *e.g.*, CLIP [17]. Blended diffusion [2] uses CLIP to compute the difference between the image embedding and the input text embedding and then injects the difference into the sampling process of a pretrained unconditional diffusion model. However, CLIP models tend to capture the global and high-level image features, thus there is no incentive to generate objects aligning with the given mask (see "Blended Diffusion" in Fig. 1). We denote this phenomenon as ***mask misalignment***. A recent GAN-based work CogNet [28] proposes to use shape information from instance segmentation dataset and predict the class of missing objects to address this problem. But it doesn't support text input. Another issue for existing inpainting methods is ***background preservation*** in which case they often produce distorted background surrounding the inpainted object as shown in Fig. 1 (bottom row).

To address above challenges, we introduce a precision factor into the input masks, *i.e.*, our model not only takes a mask as input but also information about how closely the inpainted object should follow the mask's shape. To achieve this we generate different types of masks from fine to coarse by applying Gaussian blur to accurate instance masks and use the masks and their precision type to train the guided diffusion model. With this setup, we allow users to either use coarse masks which will contain the desired object somewhere within the mask or to provide detailed masks that outline the shape of the object exactly. Thus, we can supply very accurate masks and the model will fill the entire mask with the object described by the text prompt (see the first row in Fig. 1), while, on the other hand, we can also provide very coarse masks (*e.g.*, a bounding box) and the model is free to insert the desired object within the mask area such that the object is roughly bounded by the mask.

One important characteristic, especially for coarse masks such as bounding boxes, is that we want to keep the background within the inpainted area consistent with the original image. To achieve this, we not only encourage the model to inpaint the masked region but also use a regularization loss to encourage the model to predict an instance mask of the object it is generating. At test time we replace the coarse mask with the predicted mask during sampling to preserve background as much as possible which leads to more consistent results (second row in Fig. 1).

We evaluate our model on several challenging object inpainting tasks and show that it achieves state-of-the-art results on object inpainting across several datasets and examples. Our user study shows that users prefer the outputs of our model as compared to DALLE-2 and Stable Inpainting across several axes of evaluation such as shape, text alignment, and realism. To summarize our contributions:

- We introduce a text and shape guided object inpainting diffusion model, which is conditioned on object masks of different precision, achieving a new level of control for object inpainting.
- To preserve the image background with coarse input masks, the model is trained to predict a foreground object mask during inpainting for preserving original background surrounding the synthesized object.
- We propose a multi-task training strategy by jointly training object inpainting with text-to-image generation to leverage more training data.

2. Related Work

Diffusion Models Diffusion models (DMs) [7, 24] learn the data distribution by inverting a Markov noising process, and they have gained wide attention recently due to their stability and superior performance in image synthesis as compared to GANs. Given a clean image x_0, the diffusion process adds noise to the image at each step t, obtaining a set of noisy latent x_t. Then, the model is trained to recover the clean image x_0 from x_t in the backward process. DMs have shown appealing results in different tasks, *e.g.*, unconditional image generation [7, 8, 25, 26], text-to-image generation [18–21], video generation [6], image inpainting [1, 2, 14, 16], image translation [15, 27, 30], and image editing [4, 5, 10].

Text-Guided Image Inpainting Taking advantage of the recent success of diffusion-based text-to-image generation models, an intuitive adaptation from a text-to-image generation to text-guided inpainting is to replace the pure random noise with the noisy background outside the mask region. However, this leads to strong artifacts, *e.g.*, generating partial objects or inconsistent content in the background. To address this problem, GLIDE [16] generates a random mask and then provides the masked image and mask as additions to the diffusion model, which learns to utilize the information outside of the mask region. Blended diffusion [2] en-

courages the output to align with the text prompt using the CLIP score. Repaint [14] proposes to resample in each reverse step, but it doesn't support text input. PaintbyWord [3] pairs the large-scale GAN with a full-text image retrieval network to enable multi-modal image editing. However, due to the structure of GAN, it cannot specifically modify the region given by the mask. TDANet [29] proposes a dual attention mechanism to exploit the text features about the masked region by comparing text with the corrupted image and its counterpart.

3. Preliminary: Diffusion Model

Given an input image x_0, we apply a forward diffusion Markov process to add noise to the image over a number of time steps t with scheduled variance β_t:

$$q(x_t|x_{t-1}) = \mathcal{N}\left(\sqrt{1-\beta_t}x_t, \beta_t \mathrm{I}\right) \quad (1)$$
$$q(x_{1:T}|x_0) = \prod q(x_t|x_{t-1}),$$

where T is the total number of steps. If $T \to \infty$, the output x_T will be isotropic Gaussian. The defined Markov process allows us to get x_t in a closed form

$$\begin{aligned} x_t &= \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\epsilon_{t-1} \\ &= \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \end{aligned} \quad (2)$$

where $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$, $\epsilon_t \sim \mathcal{N}(0, \mathrm{I})$.

To generate images from random noise, we need to invert above diffusion process, *i.e.*, learning $q(x_{t-1}|x_t)$ that is also a Gaussian when β_t is small enough. However, $q(x_{t-1}|x_t)$ is unknown since it is inaccessible to the true distribution of x_0. Thus, we train a neural network p_θ to approximate the conditional distribution.

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

where μ_θ is trained to predict $x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}\epsilon_t}\right)$, which is derived from Eq. (2). Since we already have x_t during training, we can train a network ϵ_θ to predict ϵ_t instead of training μ_θ [7]. We obtain the objective for training the diffusion model.

$$\mathcal{L} = \mathbb{E}_{t\sim[1,T],x_0,\epsilon_t} \|\epsilon_t - \epsilon_\theta(x_t, t)\|_2^2 \quad (4)$$

At test time, we start from a random noise $x_T \sim \mathcal{N}(0, \mathrm{I})$ and then iteratively apply the model ϵ_θ to obtain x_{t-1} from x_t until $t = 0$. We may employ more efficient sampling techniques like DDIM [25] and PNDM [13] to speed up the sampling, and adopt classifier free guidance [9] to improve the sample quality.

As for conditional diffusion models, *e.g.*, text-to-image and inpainting models, conditional information can be fed into the network ϵ_θ without changing the loss function. The model will learn to utilize the conditions to generate high quality conditional images.

4. Our Approach

Given an image x, text prompt d and a binary mask m to indicate which region of x we should modify, our goal is to generate an image $\tilde{x}$ such that the background of $\tilde{x}$ is the same as input x while the generation in the masked region $\tilde{x} \odot m$ aligns well with the text prompt d and the mask m.

4.1. Text and Shape Guided Diffusion

Existing inpainting models randomly erase part of the images and are trained to inpaint the erased region. As a result, the randomly erased region may contain only parts of an object or contain areas of background around a given object. Therefore, we propose to utilize the text and shape information from existing instance or panoptic segmentation datasets. These datasets contain annotated masks $\{m_i\}_{i=1}^N$ where N is the number of annotations and each masked region $x \odot m_i$ contains only one object. For each mask we also have a corresponding class label c_i, *e.g.*, *hat* or *cat*.

In the forward process, we randomly draw a segmentation mask m and its corresponding class text label c for image x. We define $x_0 = x$ and only add noise in the masked region instead of all pixels:

$$\begin{aligned} \tilde{x}_t &= \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon \\ x_t &= \tilde{x}_t \odot m + x_0 \odot (1-m), \end{aligned} \quad (5)$$

where $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and t is the timestep in the forward process. We use x_t, m, and c as input to the model so it can learn to utilize the clean background information and learn to recover the masked region $x_0 \odot m$. This ensures that generated objects in the foreground m are consistent with the background. Following [7] we train a network ϵ_θ to predict the noise ϵ from the noisy x_t:

$$\mathcal{L}_{\text{DM}} = \mathbb{E}_{\epsilon\sim\mathcal{N}(0,I)}\left[\|\epsilon - \epsilon_\theta(x_t, t, m, c)\|_2^2\right]. \quad (6)$$

In the inference phase, we generate random Gaussian noise in the masked region $x_T = \epsilon \odot m + x_0 \odot (1-m)$, where T is the number of sampling steps. Then we reverse the diffusion process and obtain the inpainted result x_0.

4.2. Shape Precision Control

Our training masks come from the segmentaion annotations and thus are accurate instance masks. Training the model with these masks will encourage the model to exactly follow the shape of the input mask at test time. To allow users to provide masks that are either accurate (e.g., in the shape of a cat) or coarse (e.g., a bounding box) we propose to generate masks with different precision. To achieve this, we randomly augment the masks during training to degrade the shape of the original mask. Specifically, given an accurate instance mask m, we use a mask precision indicator $s \sim [0, S]$ and define a set of parameters for each indicator:

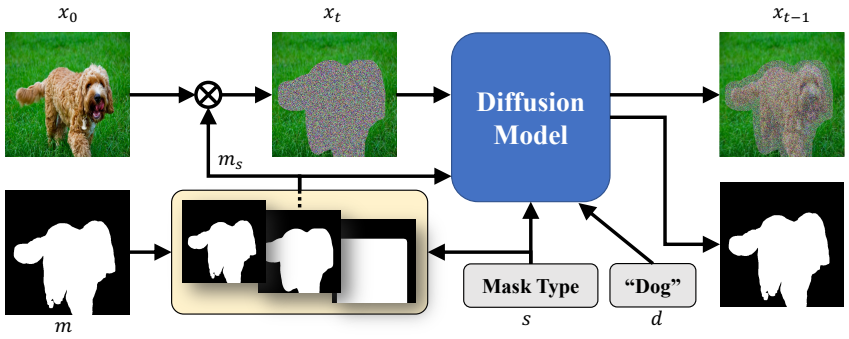

Figure 2. Text and shape guided object inpainting. Given an image x_0, accurate mask m and object description d, we transform the mask m to different precision levels (from accurate to coarse) as m_s. We add noise in the masked region to provide rich background information to the diffusion model and train the model to predict the added noise as well as the accurate mask m. During inference, we apply the diffusion model repeatedly until $t = 0$.

$$m_s = \text{GaussianBlur}(m, k_s, \sigma_s), \tag{7}$$

where k_s denotes Gaussian kernel size, and σ_s is standard deviation of the kernel. If $s = 0$, the mask stays unchanged and corresponds to the accurate instance mask from the dataset annotation. When $s = S$, the mask m_s is a bounding box of the instance mask m, and it loses all detailed shape information. During training, for each training sample (object), we employ a set of masks $\{m_s, s\}$ from fine to coarse and condition the diffusion model on the precision indicator s:

$$\mathcal{L}_{\text{seg-DM}} = \mathbb{E}_{\epsilon\sim\mathcal{N}(0,I)}\left[\|\epsilon - \epsilon_\theta(x_t, t, m_s, c, s)\|_2^2\right]. \tag{8}$$

Through this, we can control whether the generated object should align with the input mask by specifying different mask precision indicators s. We present a sample of masks in Fig. 5.

4.3. Background Preservation

During inference, the diffusion model will denoise the masked region and generate objects according to the given text prompt. As a result, the background in the masked region will be changed if the input masks are coarse. For example, the model may generate a cat in the given square box mask region but the other pixels in the square box region will also be changed. Ideally we would like to preserve the background, however, this is challenging since we do not know where in the coarse mask the model will generate the desired object.

We address this challenge by utilizing the information of mask precision. Specifically, we train our diffusion network to also predict an accurate instance mask m from the coarse input version m_s:

$$\mathcal{L}_{\text{prediction}} = H(\epsilon_\theta(m_s), m), \tag{9}$$

where H can be any suitable criterion for segmentation. We choose to use the DICE loss, *i.e.*, $H(X, Y) = 1 - \frac{2|X\cap Y|}{|X|+|Y|}$. For this, we simply add an extra output channel to our diffusion model which contains the instance mask prediction.

During inference, we are able to predict where the object is generated inside the coarse mask m_s using the diffusion model's prediction. We first feed a coarse mask m_s into the diffusion model and switch to using the predicted mask to perform denoising. With the predicted mask, we know where the object is generated within the masked region which helps to preserve background information around the generated object.

4.4. Training Strategy

Combining Eqs. (8) and (9), our final training objective can be expressed as follows.

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{seg-DM}} + \lambda\mathcal{L}_{\text{prediction}}, \tag{10}$$

where λ is a hyper-parameter which balances the two losses. In our experiment, $\lambda = 0.01$.

Our model can be built based on pre-trained text-to-image generation models, *e.g.*, Stable Diffusion and Imagen, to speed up the training process. In the experiments, we finetune based on the Stable Diffusion text-to-image model v1.2 with our conditions (Fig. 2) and loss function $\mathcal{L}_{\text{total}}$ (Eq. (10)). To align text descriptions with the local mask content, avoiding text misalignment as aforementioned, we train with the training split of OpenImages v6 , which has segmentation and corresponding labels that can serve as local descriptions. From our empirical study, such categorical text would degrade the generation quality from long sentences. Therefore, we employ the BLIP model [11] to collect richer and longer captions for those local segments.

https://github.com/CompVis/stable-diffusion
https://storage.googleapis.com/openimages/web/index.html

During the training, we randomly pair the segmentation label or BLIP caption to the corresponding mask. Therefore, the model can handle both single word text and short phrase well during the inference.

Multi-task Training To leverage more training data and handle more diverse text descriptions and image contents, beyond the domain of the segmentation dataset, we propose a multi-task training strategy by jointly training our main task and the foundational text-to-image generation task, using image/text paired data from LAION-Aesthetics v2 5+ subset [22] following Stable Diffusion [20]. For text-to-image, we set the input mask to cover the entire image, and treat it as a special inpainting case. As demonstrated in Sec. 5, our final model trained with all these components significantly outperforms state-of-the-art methods in terms of visual quality of generated objects, as well as their consistency to text description and mask shape.

5. Experimental Evaluation

5.1. Experimental Setup

We set $\lambda = 0.01$ in the total loss function Eq. (10) and batch size to be 1024. Following the training strategy discussed in Sec. 4.4, we train the inpainting task and text-to-image generation task with the probability of 80% and 20%, respectively. Our model was trained around 20K steps on 8 A100 GPUs. As a reference, Stable Inpainting takes 256 A100 GPUs around 440K steps.

Baselines We choose the state-of-the-art image inpainting methods as our baselines, *i.e.*, Blended Diffusion [2], GLIDE [16], Stable Diffusion [20], and Stable Inpainting [20]. We also compare with DALLE-2 [18] on limited images since its model is not open source yet. Stable Diffusion, Stable Inpainting, and our SmartBrush support image generation on the size of 512×512. Since Blended Diffusion and GLIDE only support images size of 256×256, we resize all results to 256×256 for fair comparison.

Testing Datasets We evaluate our model on two popular segmentation datasets, *i.e.*, OpenImages [22] and MSCOCO [12]. We sample 2 masks for each image in the testing dataset of MSCOCO, so the number of testing images is 9311. As for OpenImages, we sample images with resolution higher than 512 and use one mask for each image. Then, the number of testing images is 13400. The input prompts are directly from segmentation class labels.

Evaluation Metrics We first measure the image quality by Frechet Inception Distance (FID) [23]. Since our main task is object generation in the masked region, the global FID cannot well reflect the generation quality since the masked region may occupy a small part of the image. Therefore, we crop the images according to the bounding box of the mask and measure FID on the local regions, which is referred to as "Local FID". To measure the alignment between text and generated content, we adopt the CLIP score [17].

5.2. Text and Shape Guided Inpainting

The proposed SmartBrush can inpaint not only objects but also generic scene like sunset sky by following the text and shape guidance. For object inpainting, we consider two common use cases: 1) accurate object masks and 2) bounding box masks. The former expects the generated object to follow the given mask shape, while the latter does not constrain the shape of generated objects as long as they are inside of the box. Corresponding quantitative results are listed in Tabs. 1 and 2. Our SmartBrush achieves the best performance in both tasks on all metrics, which demonstrates the effectiveness of our proposed training strategy with text and shape guidance.

Fig. 3 visualizes inpainting examples from the baselines and our SmartBrush. In general, we can generate high-quality objects/scenes well following both the mask shape and text, no matter short words or long sentences. By contrast, all baselines failed following the mask shape. Besides object inpainting, our SmartBrush also supports scene inpainting as illustrated by the last two rows in Fig. 3. More examples can be found in the supplementary. Still, as compared to our SmartBrush, it is difficult for existing inpainting models to follow the mask shape.

We also conduct user studies through Amazon Mechanical Turk. Over 300 workers were asked 1) which result follows the object mask best, 2) which result follows the input text description best, and 3) which result looks most natural/realistic. The survey result is shown in Fig. 4, where more than 50% users vote our results as the best on each question.

5.3. Mask Precision Control

In the real world, users will not always provide the precise mask of the object they want to inpaint. We may encounter a coarse mask, so SmartBrush accepts the control of how closely the inpainted object is to the given mask. Fig. 5 shows the results with different types of masks, which follow the blurring rule during training, *i.e.*, applying Gaussian blur iteratively to obtain masks from fine to coarse. The Stable Diffusion results are not affected by mask types since it is not trained that way. The results of Stable Inpainting only change the object size with the mask size but do not follow the mask shape. By contrast, ours strictly follow the mask shape when providing a finer mask, while roughly following the mask if given a coarser mask. For extremely, given a box-like mask (the last column), we allow the generation to happen anywhere inside the box.

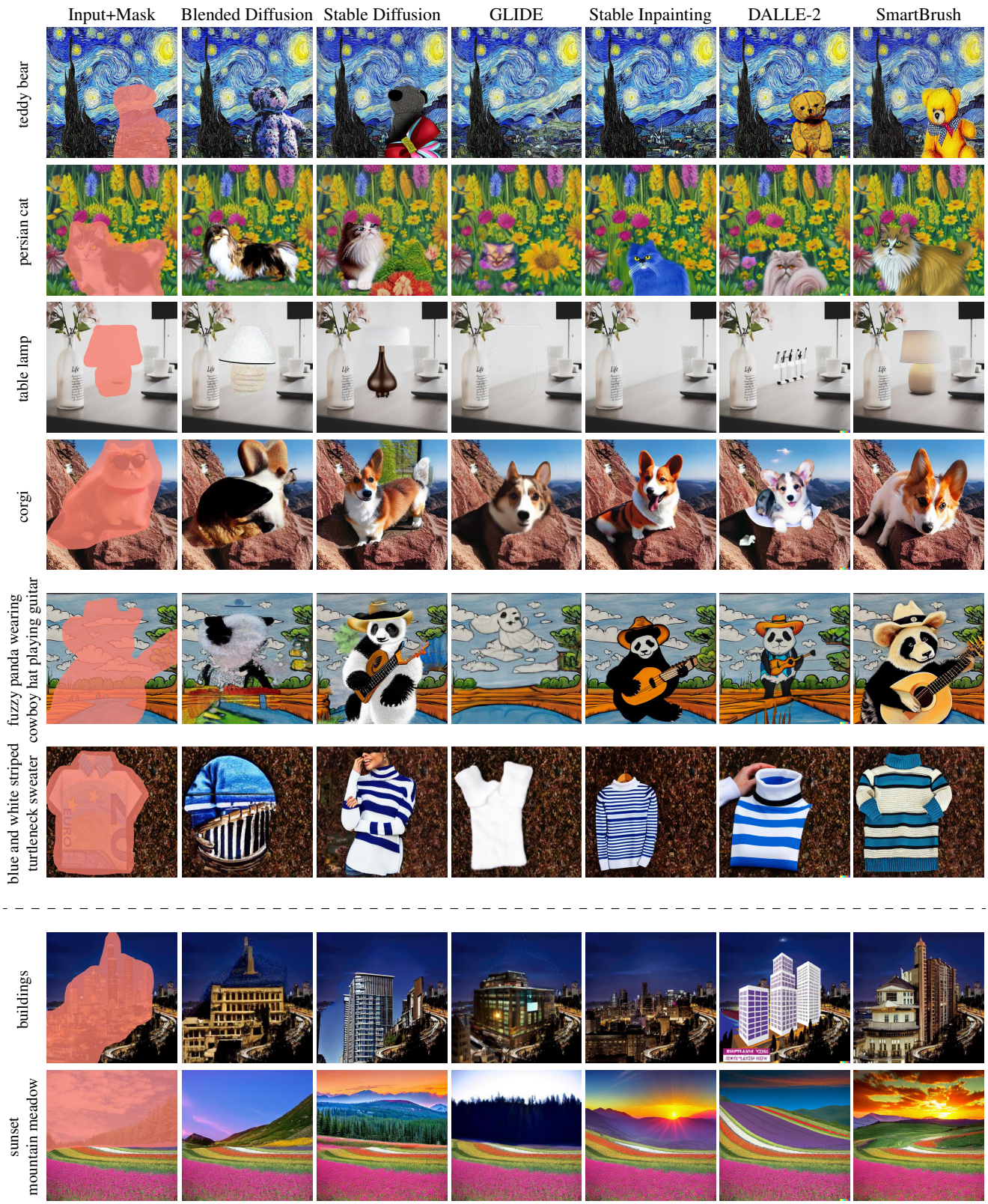

Figure 3. Comparison of text and shape guided inpainting.

Table 1. Text-guided object inpainting with bounding box mask.

	OpenImages			MSCOCO		
	Local FID ↓	CLIP Score ↑	FID ↓	Local FID ↓	CLIP Score ↑	FID ↓
Blended Diffusion [2]	29.16	0.265	11.05	41.43	0.251	12.68
GLIDE [16]	22.45	0.252	9.70	30.72	0.241	9.32
Stable Diffusion [20]	15.28	0.265	9.10	25.61	0.250	12.29
Stable Inpainting [20]	12.57	0.264	7.07	18.13	0.246	8.50
SmartBrush (Ours)	**9.71**	**0.266**	**6.00**	**13.22**	**0.252**	**8.05**

Table 2. Text-guided object inpainting with object layout mask.

	OpenImages			MSCOCO		
	Local FID ↓	CLIP Score ↑	FID ↓	Local FID ↓	CLIP Score ↑	FID ↓
Blended Diffusion [2]	21.93	0.261	9.72	26.25	0.244	8.16
GLIDE [16]	21.09	0.250	9.03	24.25	0.235	6.98
Stable Diffusion [20]	12.27	**0.263**	6.90	17.16	0.246	7.78
Stable Inpainting [20]	10.98	0.261	5.84	15.16	0.243	6.54
SmartBrush (Ours)	**7.82**	**0.263**	**4.70**	**9.80**	**0.249**	**5.76**

Method	**LFID↓**	**CLIP↑**	**FID ↓**
Ours	13.22	0.252	8.05
+ Background Preservation	12.26	0.251	7.19
- Mask Precision Cond	15.31	0.252	8.57
- BLIP Prompts	13.52	0.249	10.69
- Multi-Task	15.26	0.250	8.26
Stable Inpainting (SOTA)	18.13	0.246	8.50
+ Finetune on Our Dataset	18.34	0.245	8.38

Table 3. Quantitative ablation study on MSCOCO dataset.

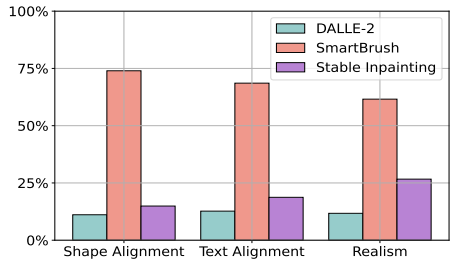

Figure 4. We ask users to choose the generation that best aligns with the mask and input text, and looks most realistic. Our method SmartBrush outperforms the baselines by a large margin.

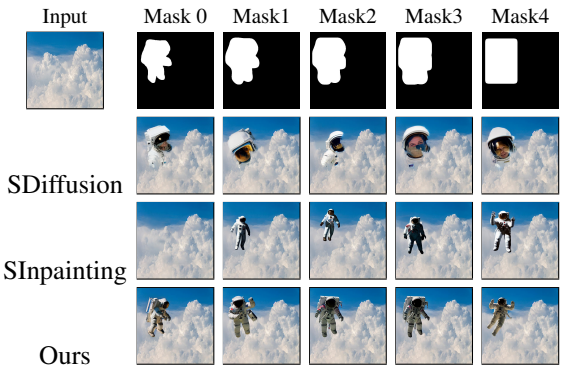

Figure 5. Mask precision control samples with prompt "*astronaut*". As we increase the mask type, our method give more freedom to the model and the outputs gradually become different from the input object shape mask.

5.4. Background Preservation

To inpaint an object, especially when giving a box-like mask, it is important to preserve the background since the inpainted object will only partially occupy the mask area. Fig. 6 compares different methods in background preservation when giving box-like masks. Without any background preservation regularization, DALLE-2 generates objects inside the mask and changes the non-object pixels inside the mask. Our SmartBrush, with object mask prediction (shown in Fig. 7), could much better preserve the background by utilizing the predicted mask during sampling.

5.5. Ablation Study

We remove the proposed component separately and test the results on MSCOCO dataset with bounding box mask. The main results are listed in Tab. 3. We didn't apply background preservation in Table .1 as it is an optional function for users. We observe that the proposed background preser-

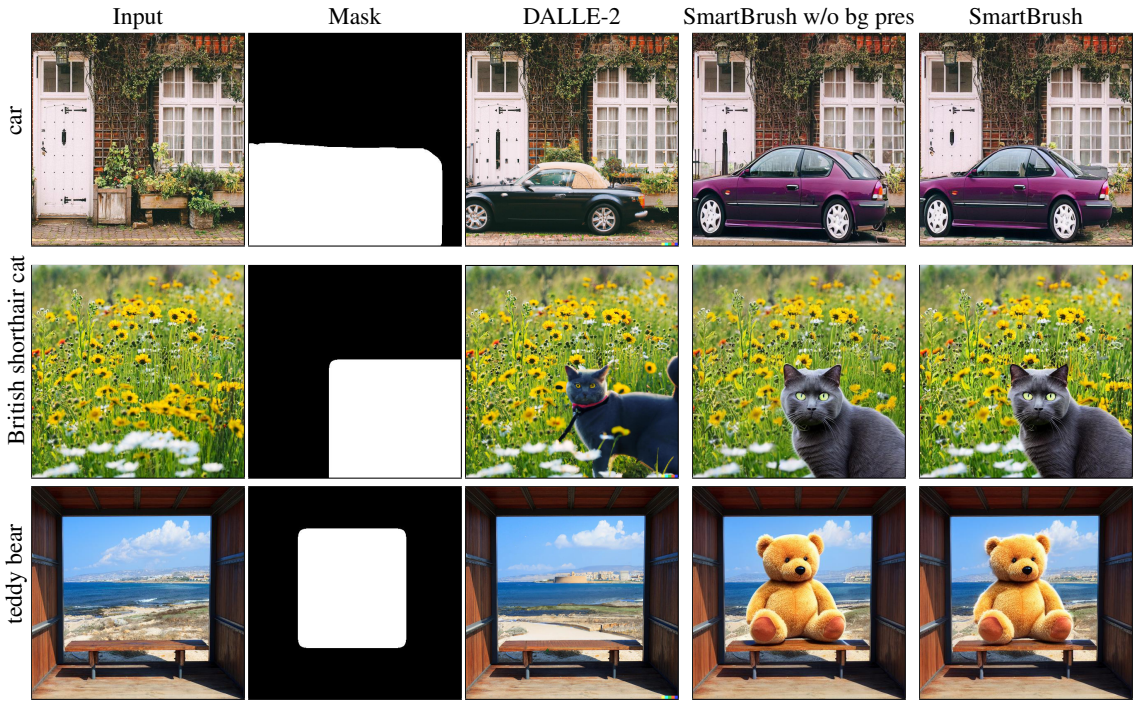

Figure 6. Comparison of background preservation in inpainting. We only compare with DALLE-2 here for better visualization, and more baseline results are provided in the supplementary. We observe that DALLE-2 and SmartBrush w/o background preservation change the background surrounding the generated object, *e.g.*, the door bell, clouds behind the mountain, flowers behind the cat, and landscape behind the teddy bear. By contrast, our SmartBrush better preserves the background pixels.

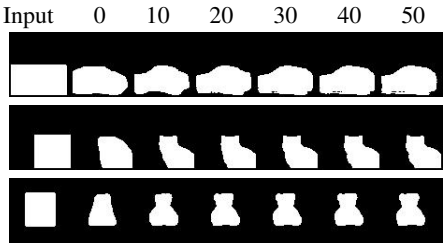

Figure 7. Predicted object masks corresponding to examples from Fig. 6. The numbers denote the sampling time steps. The mask prediction becomes sharper after around 10 steps.

vation improves the quantitative results as more pixels are preserved. If we don't apply mask precision conditioning, the generated objects are not well controlled and lead to bad Local FID (LFID). The degraded performance without BLIP and Multi-task training also demonstrate that they are useful for the object inpainting task.

6. Conclusion, Limitation, and Future Work

Existing text and shape guided image inpainting models face three typical challenges: mask misalignment, text misalignment, and background preservation. In this paper, we propose a novel training method that utilizes the text and shape guidance from the segmentation dataset to address the text misalignment problem. Then we further propose to create different levels of masks (from fine to coarse) to allow precision control of the generation. Finally, we propose an additional training loss function to encourage the model to make object predictions from the input box mask. Then we can utilize the predicted mask to avoid unnecessary changes inside the mask. The quantitative and qualitative results demonstrate the superiority of our method.

The main limitation of our method is the large shadow case, where the shadow of the object exceeds the object mask, *e.g.*, the shadow of a person can be very long in the morning while the bounding box usually fails to cover the whole shadow. Our method may not be able to generate such long shadow since the coarsest mask is the object bounding box. We will explore it in the near future.

Acknowledgement

This project was partially supported by the National Institutes of Health (NIH) under Contract R01HL159805, by the NSF-Convergence Accelerator Track-D award 2134901, by a grant from Apple Inc., a grant from KDDI Research Inc, and generous gifts from Salesforce Inc., Microsoft Research, and Amazon Research.

References

[1] Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *arXiv preprint arXiv:2206.02779*, 2022. 2

[2] 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. 2, 5, 7

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

[4] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022. 2

[5] 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

[6] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 2

[7] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1, 2, 3

[8] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23:47–1, 2022. 2

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

[10] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. *arXiv preprint arXiv:2210.09276*, 2022. 2

[11] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *arXiv preprint arXiv:2201.12086*, 2022. 4

[12] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 5

[13] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022. 3

[14] 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*, pages 11461–11471, 2022. 2, 3

[15] 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*, 2021. 2

[16] 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. 2, 5, 7

[17] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 2, 5

[18] 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, 5

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

[20] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1, 2, 5, 7

[21] 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

[22] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 5

[23] Maximilian Seitzer. pytorch-fid: FID Score for PyTorch. https://github.com/mseitzer/pytorch-fid, August 2020. Version 0.2.1. 5

[24] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 1, 2

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

[26] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[27] Tengfei Wang, Ting Zhang, Bo Zhang, Hao Ouyang, Dong Chen, Qifeng Chen, and Fang Wen. Pretraining is all you need for image-to-image translation. *arXiv preprint arXiv:2205.12952*, 2022. 2

[28] Yu Zeng, Zhe Lin, and Vishal M Patel. Shape-guided object inpainting. *arXiv preprint arXiv:2204.07845*, 2022. 2

[29] Lisai Zhang, Qingcai Chen, Baotian Hu, and Shuoran Jiang. Text-guided neural image inpainting. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1302–1310, 2020. 3

[30] Min Zhao, Fan Bao, Chongxuan Li, and Jun Zhu. Egsde: Unpaired image-to-image translation via energy-guided stochastic differential equations. *arXiv preprint arXiv:2207.06635*, 2022. 2