Uni-paint: A Unified Framework for Multimodal Image Inpainting with Pretrained Diffusion Model

Shiyuan Yang
City University of Hong Kong
Hong Kong SAR, China

Xiaodong Chen
Tianjin University
Tianjin, China

Jing Liao*
City University of Hong Kong
Hong Kong SAR, China

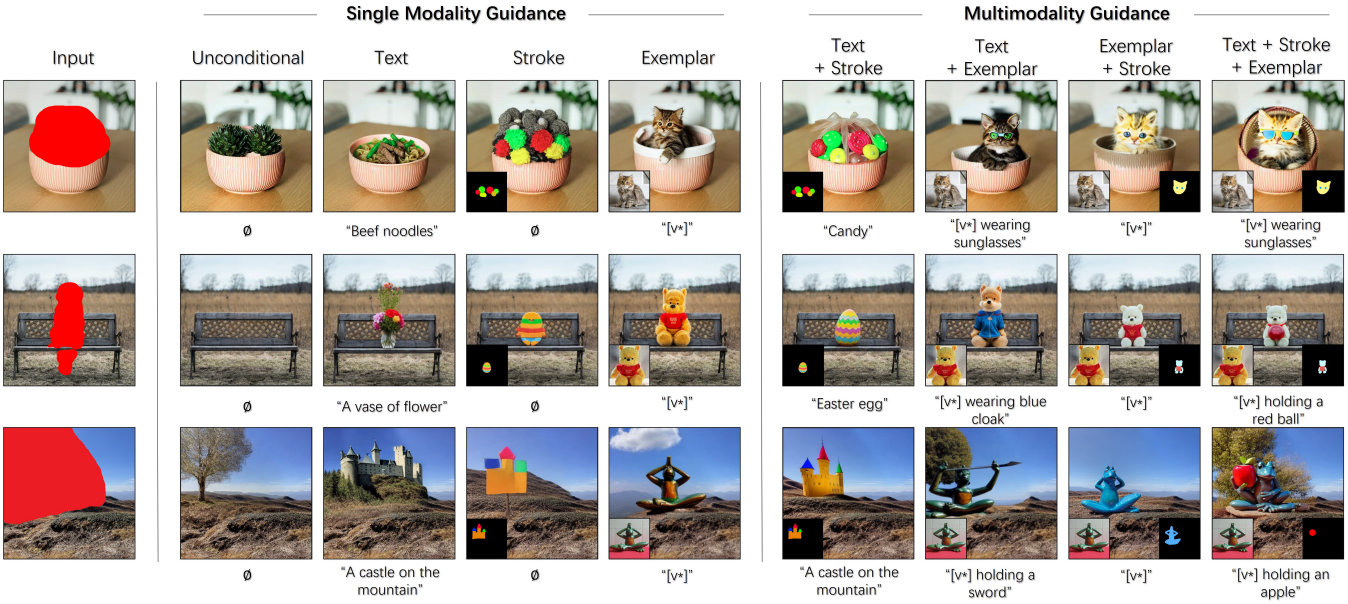

Figure 1: Uni-paint allows users to perform unconditional, text-guided, stroke-guided, exemplar-guided or mix-guided inpainting on a single provided image within one unified framework.

ABSTRACT

Recently, text-to-image denoising diffusion probabilistic models (DDPMs) have demonstrated impressive image generation capabilities and have also been successfully applied to image inpainting. However, in practice, users often require more control over the inpainting process beyond textual guidance, especially when they want to composite objects with customized appearance, color, shape, and layout. Unfortunately, existing diffusion-based inpainting methods are limited to single-modal guidance and require task-specific training, hindering their cross-modal scalability. To address these limitations, we propose Uni-paint, a unified framework for multimodal inpainting that offers various modes of guidance, including unconditional, text-driven, stroke-driven, exemplar-driven inpainting, as well as a combination of these modes. Furthermore, our Uni-paint is based on pretrained Stable Diffusion and does not require task-specific training on specific datasets, enabling few-shot generalizability to customized images. We have conducted extensive qualitative and quantitative evaluations that show our approach achieves comparable results to existing single-modal methods while offering multimodal inpainting capabilities not available in other methods. Code will be available at https://github.com/ysy31415/unipaint.

1 INTRODUCTION

Large-scale language-image models such as denoising diffusion probabilistic models (DDPMs) [12, 31, 32, 35] have recently shown impressive generation quality and domain generizibility surpassing that of GANs. As a promising generative modeling paradigm, diffusion models have also been applied to image inpainting. Current diffusion-based inpainting methods can be divided into two categories: training-based and few-shot methods. Training-based methods involve either training an image-to-image diffusion model [34] or modifying a pretrained text-to-image model [32, 43, 45, 48] with additional conditioning (e.g., masked image). While these models have fast inference times, they suffer from several drawbacks. Firstly, large-scale dataset acquisition can be challenging. Secondly, these methods are less scalable for modal extension as they have been specifically designed and trained for a certain modality. On the other hand, few-shot methods directly leverage the powerful generative capability of an off-the-shelf pretrained model through model prior [1, 24] or guided sampling [2], requiring no additional dataset collection and training. This category offers higher flexibility and scalability, making it preferred in our work.

*Corresponding author.

Despite the success of diffusion models in the inpainting task, their potential has not been fully exploited yet. Current methods, regardless of their category, have limited capabilities for modal extension, supporting only unconditional inpainting [24] or single modality guidance (text guidance [1, 32, 48], or exemplar-guidance [47]). This lack of flexibility in general usage can be problematic, as a combination of different interaction methods is often required to achieve a satisfactory inpainting result. For example, while textual descriptions can be used to describe high-level semantics to be inpainted, they may struggle to accurately convey the user's intentions for object shape, color, and customized attributes. This can be addressed by providing an additional reference image (exemplar) or drawing rough color strokes. As shown by the cat at upper right corner in Fig. 1, generating such cat with specific identity, predetermined colors, and gestures is much easier using multiple guidance rather than relying solely on text. Therefore, a framework that enables multimodal conditions for image inpainting is a natural choice, but existing methods do not support it.

In this work, we present Uni-paint, the first unified framework for multimodal image inpainting that supports both unconditional and conditional controls, including text, stroke, exemplar, and a combination of them, as shown in Fig. 1. To achieve this, we first finetune a pretrained Stable Diffusion model [32] unconditionally, requiring it to generate images that are only faithful to the known part of the input image, which we refer to as masked finetuning. Since the Stable Diffusion has been extensively pretrained on large image datasets, it possesses the prior knowledge needed to generate plausible images. Our masked finetuning further enables the model to generate context-plausible content in the unknown region unconditionally by leveraging its learned semantic awareness of the known part. Furthermore, by exploiting the existing conditional interface of a text-to-image diffusion model, conditional inpainting with multiple modalities is also unified in this framework. We identify two types of conditional interfaces: (1) the textual interface, implemented through cross-attention, applicable for semantic guidance like text and exemplar, and (2) the spatial interface, achieved through image blending, suitable for spatial guidance like stroke. These guidance modes can even be combined in the same framework to perform mixed-modal inpainting.

就是 stable inpainting 在做的事

Our Uni-paint is a few-shot method that differs from previous approaches [32, 34, 48] that require training the model on large datasets. Our method only requires finetuning on a single input image, reducing the dependency on data collection and eliminating restrictions to training domain. However, like other few-shot inpainting methods [1, 2, 24], our approach needs to progressively blend the inpainted content with the known regions of the input image during the sampling process to keep the known region untouched. A common issue in blending-based methods is that the inpainted content may overflow the mask boundary and get truncated after blending. To address this, we introduce a masked attention control mechanism for cross-attention and self-attention layers of the diffusion model to restrict the scope of the generated content within the unknown area.

few-shot 難的是沒有看過其他 modal (text, storke...)跟圖像的對應關係

Our Uni-paint framework has undergone extensive qualitative and quantitative evaluations that demonstrate its comparable results to existing single-modal methods while offering multimodal inpainting capabilities that are not available in other methods. In summary, our contributions are as follows:

- We propose the first unified framework for multimodal image inpainting based on pretrained diffusion model.
- We introduce few-shot masked finetuning on a single image with null conditioning, making the inpainting scalable to other modalities and generalizable to customized image inputs.
- We introduce a masked attention mechanism to alleviate the potential leakage of inpainted content to known areas.

2 RELATED WORK

Image inpainting. Early methods relied on borrowing the low-level texture patches from known regions [3, 6, 9], but struggled with complex semantic scenes like face completion. This problem was not solved until ContextEncoder [29], the first GAN-based model was proposed. Subsequent CNN/GAN-based works achieved improved results by incorporating various modules like Partial Convolution [22], Contextual Attention [49], Fourier Convolution [38], etc. Others works used multi-stage pipelines like edge-guided [27, 46], coarse-to-fine [37, 42, 50], progressive [20, 51], and recurrent [21] networks. The recent RePaint method [24] leveraged DDPM priors and repetitive sampling for promising results. However, these unconditional methods do not support user guidance.

Text-driven image editing. As a user-friendly guidance modality, text-driven editing has been gaining popularity. Early GAN-based method like StyleCLIP [28] achieved human face editing by leveraging pretrained StyleGAN [15] and CLIP [30]. Recently, the rise of large text-to-image diffusion models [31, 32, 35, 48] has paved a promising way for high-quality, high-diversity text-driven generative modeling. These works can be roughly divided into three categories: (1) Guided Sampling: by introducing various guidance functions during sampling, such as CLIP guidance [2], style guidance [23], edge guidance [41], and attention guidance [5]. (2) Attention Control: Works like Prompt-to-Prompt[10] and shape-guided editing [14] manipulate the attention layer for impressive editing results. (3) Training/Finetuning: InstructPix2Pix [4] trained a model on generated image-prompt pairs for fast editing with user instructions. Few-shot editing can also be achieved by finetuning the model [17, 40], optimizing the text embedding [7, 26], or both [16]. Text-driven image editing with diffusion models has also been applied to image inpainting by altering a pretrained text-to-image model to enable masked image conditioning [32, 43, 45, 48], but these methods require large training datasets. Alternatively, Blended Diffusion [1, 2] achieved zero-shot inpainting by utilizing background blending. However, the blending strategy does not expose the model to full context and may fail to handle semantic transitions near the hole boundary. In contrast, our method employs masked finetuning and attention control, ensuring global context awareness and improved texture transitions.

Exemplar-driven image editing. Exemplar-driven image editing is a relatively new topic, which allows users to synthesize customized objects using provided exemplars. Textual inversion [7] captures new concepts from exemplars by optimizing token embedding. DreamBooth [33] generates personalized outputs by finetuning the model on exemplar images. Follow-up works like CustomDiff [19], SVDiff [8], and ELITE [44] further accelerate the

process by finetuning the model's cross-attention layer, singular values of the weights, and learning a mapping encoder, respectively. Paint-by-Example [47] first achieved exemplar-driven inpainting by trading textual conditioning for image conditioning, leading to a lack of text guidance support. Also, its generalization to unseen objects may also be limited by the training dataset. Our approach mitigates these issues by extending a pretrained text-to-image model for user-specific sample finetuning without modifying the model structure.

3 METHOD

In our inpainting task, given an incomplete input image x^{in} with a binary mask m indicating its known region. Our uni-paint aims to inpaint its unknown part and outputs inpainted image x^{out} unconditionally or conditioned on multimodal guidance, including text prompt w, exemplar image x^{ref}, stroke map x^{stk}, or even a mix of them, all based on a model parameterized by θ. This high-level process can be formulated as:

$$x^{out} = \text{Unipaint}_\theta(x^{in}, m, [w, x^{ref}, x^{stk}]), \tag{1}$$

where $[\cdot]$ denotes optional conditional inputs. Technically, we implemented it through an iterative deterministic DDIM denoise sampling [36] based on Stable Diffusion [32], where each denoise step $p(x_{t-1}|x_t, c)$ is formulated as:

deterministic DDIM 表示在 DDIM過程中沒有隨機項。

$$x_{t-1} = \sqrt{\alpha_{t-1}}\left(\frac{x_t - \sqrt{1-\alpha_t}\epsilon_\theta(x_t, c, t)}{\sqrt{\alpha_t}}\right) + \sqrt{1-\alpha_{t-1}}\cdot\epsilon_\theta(x_t, c, t), \tag{2}$$

where α_t is time-dependent hyperparameter, ϵ_θ is the noise predictor (i.e., the model) which takes current sampled image x_t, text embedding c, timestep t as inputs, and predicts the noise for denoising sampling. The initial image x_T is sampled from $\mathcal{N}(\mathbf{0}, \mathbf{I})$ and the final denoised image x_0 will act as the output x^{out}.

3.1 Overview

To leverage such model ϵ_θ for inpainting, first we need to preserve the known region of x^{out} as in x^{in}. A simple way is to blend the known part with x_t during sampling as done in [1, 2]. However, we found this is insufficient since the known information is inserted externally rather than generated by the model itself, the model lacks full context awareness, potentially causing incoherent semantic transitions near hole boundary. As such, we additionally finetune the model weights $\theta \to \theta^*$ such that it can inherently reconstruct the known region, as shown on left side of Fig. 2. We demonstrate its superiority over direct blending in ablation study.

The finetuned model ϵ_{θ^*} is aware of the known region. To achieve multimodal inpainting in unknown region, according to Eq. 2, we can alter the sampling by injecting semantic guidance via c, or spatial guidance via x_t, or not apply any guidance, depending on specific needs, as summarized on the right side of Fig. 2. We inject the conditional information accordingly:

(1) Unconditional inpainting can be achieved by leveraging model's own prior without any conditions (c is set to null text $\emptyset$), formulated as $\epsilon_{\theta^*}(x_t, \emptyset, t)$, as illustrated in Uncond. block in Fig. 2.

(2) Given text w, we obtain its embedding c through a text encoder C, the sampling is conditioned on c via semantic interface, formulated as $\epsilon_{\theta^*}(x_t, C(w), t)$ as shown by Text block in Fig. 2.

(3) Given exemplar x^{ref}, despite being in image format, it is fundamentally different image composition. We expect the model to represent its distinct semantic features with some variation rather than naive copy-and-paste. Therefore, we associate x^{ref} with an auto-selected token v^* and inject it through semantic interface c, formulated as $\epsilon_{\theta^*}(x_t, C(v^*), t)$, as shown by Exemplar block.

(4) Given stroke map x^{stk}, since it only requires color and spatial alignment while permitting ambiguous semantic interpretation, and text is struggled to deliver spatial information, so the best choice is to spatially inject x^{stk} into x_t to obtain modified x_t' (see Stroke block). The model operates as $\epsilon_{\theta^*}(x_t', \emptyset, t)$.

These modalities can be used individually for single modality inpainting, or combined for multimodality inpainting, which we will introduce next.

3.2 Unconditional inpainting

We begin with unconditional inpainting as it serves as the foundation for conditional guidance.

Masked Finetuning. Stable Diffusion has a strong generative prior learned from extensive pretraining, we leverage such prior for unconditional completion by introducing masked finetuning, where the model is finetuned to reconstruct the known part of the image. The masked finetuning enables the model to leverage its learned semantic understanding of the known areas, resulting in a plausible completion in unknown region.

Specifically, we adopt a typical training scheme as used in stable diffusion [32], which reduces the computational load by first having an encoder map the input image to low-dimension latent map (for the rest of this paper, notation x refers to latent representation). The noising-denoising process is then performed in latent space, supervised by a simple noise loss derived in DDPM [12]. The difference is that we only calculate the loss on known region (specified by m). Our masked finetuning loss ℓ_{bg} is defined below:

$$\ell_{bg} = \mathbb{E}_{\epsilon,t}\left\|m \odot \epsilon - m \odot \epsilon_\theta\left(x_t^{in}, \emptyset, t\right)\right\|_2^2, \tag{3}$$

where x_t^{in} is noised x^{in} at timestep t: $x_t^{in} = \sqrt{\alpha_t}x^{in} + \sqrt{1-\alpha_t}\epsilon$ ($\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$), ϵ_θ is the predicted noise from the model, $\emptyset$ is the null text embedding. Even if there is no explicit constraint on unknown area, the model is still able to complete missing area in the sampling stage. This can be attributed to the natural coherence of the pre-training images, as well as the inherent inductive bias of convolution and self-attention layers to extend and replicate textures.

Sampling with blending.

While finetuning the model helps memorize semantic content in known regions, yet does not guarantee perfect reconstruction, unless we overfit the known region through excessive finetuning, which can negatively impact the model's generative ability. To preserve the known region and the editability, we also employ blending technique, where the known part of sampled latent x_t is replaced by noised x^{in} after each sampling step. This helps preserve the known region with much fewer iterations and less tuning on weights.

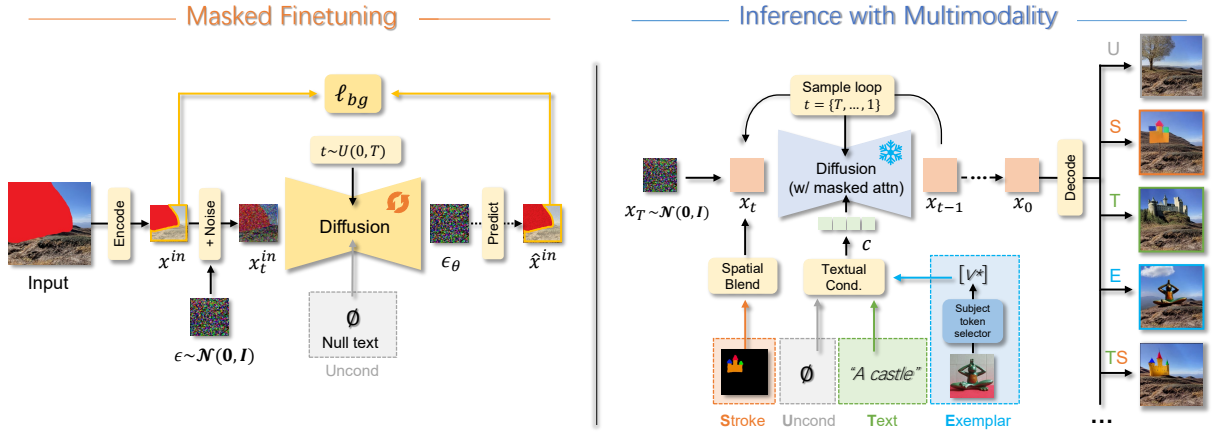

Figure 2: Pipeline overview of Uni-paint. The model is finetuned on the known area of the input with null text. During sampling, unconditional, text-driven, stroke-driven and exemplar-driven inpainting can be achieved by conditioning on null text, text, stroke map, and exemplar's subject token, respectively.

3.3 Text-driven inpainting

Once the model has been finetuned on known region, it will be immediately available for text-driven inpainting by conditioning on a text prompt w. Classifier-free guidance [13] is also applied to boost the fidelity:

$$\hat{\epsilon}_{\theta^*}(x_t, c, t) = \epsilon_{\theta^*}(x_t, \emptyset, t) + s\left(\epsilon_{\theta^*}(x_t, c, t) - \epsilon_{\theta^*}(x_t, \emptyset, t)\right), \quad (4)$$

where $c = C(w)$ is the embedding of w, s is the guidance scale. Since we do not apply constraint on unknown region when finetuning, thus the model tends to respond to text roughly in the unknown area. Applying masked attention can bring more accurate control to editing scope, which will be discussed later.

3.4 Exemplar-driven inpainting

Exemplar-driven inpainting allows the user to provide an exemplar image x^{ref} containing a subject, and the model should inpaint the subject in the unknown region while maintaining a reasonable semantic relationship with the background. Unlike image composition, the inserted subject should have the same identity as the exemplar but with variations rather than a simple copy-and-paste. Therefore, this is more challenging as it requires the model to remember exemplar's key semantic features.

To achieve this, we finetune the model on exemplar conditioned a class token v^* that roughly corresponds to the subject. This process is performed in parallel with the unconditional finetuning by appending the following reference loss ℓ_{ref}:

$$\ell_{ref} = \mathbb{E}_{\epsilon,t}\left\|\epsilon - \epsilon_\theta\left(x_t^{ref}, C(v^*), t\right)\right\|_2^2, \quad (5)$$

where x_t^{ref} is noised exemplar at timestep t, $C(v^*)$ returns the embedding of v^*. We do not include prior-preservation loss as done in DreamBooth [33] since this can be unnecessary in our task, if users want to generate non-customized instances of a class, text-driven inpainting is a better alternative. To accelerate convergence and bring more diversity to object scale and position, we apply the augmentation by randomly scaling and shifting the exemplar image

unknow bounding box, valid region 都
是指圖像中需要生成的區域的範圍。

inside the unknown area's bounding box, and only calculate the loss on valid region. During the inference, conditioning on $C(v^*)$ allows the model to represent the subject.

Automatic subject identification. In normal penalization works [7, 33], users first need to manually specify an initial subject token that roughly describes the exemplar. Here we provide an alternative to automatically obtain the subject token v^*, which can be useful when users are unsure about the subject category or in automated scenarios. Since Stable Diffusion uses CLIP text model [30] as its text encoder, we use the corresponding CLIP image encoder to retrieve the token v^* that best matches the exemplar. We represent this process as follows:

$$v^* = \arg\max_{v_i \in \mathcal{V}}\left(E_T(v_i) \cdot E_I(x^{ref})\right), \quad (6)$$

where $\mathcal{V}$ is the set of all tokens, $E_T(v_i)$ is the CLIP text embedding of v_i, $E_I(x^{ref})$ is the image embedding of x^{ref}. In practice, embedding set $\{E_T(v_i)\}$ can be pre-computed and stored, only a single inference of $E_I(x^{ref})$ is needed which costs negligible of time. We visualize such process in appendix Sec. A.2.

3.5 Stroke-driven inpainting

Stroke-driven inpainting aims to generate real objects in unknown area that have a consistent shape and color with the user's stroke map. A finetuned model is also immediately available for stroke inpainting. To achieve this, we spatially blend the stroke latent x^{stk} with sampled latent x_t to obtain modified latent x_t' at a certain intermediate timestep τ during the sampling, the blending operation B is formulated as follows:

$$x_t' = B(x^{stk}, x_t, t) = \begin{cases} x_t \odot (1 - m^{stk}) + x_t^{stk} \odot m^{stk} & \text{if } t = \tau \\ x_t & \text{otherwise} \end{cases} \quad (7)$$

where x_t^{stk} is noised stroke latent at time t, m^{stk} is stroke mask derived from x^{stk}. It is noteworthy that our approach does not require users to scribble over the entire unknown area, but only the

desired object, surrounding region will be completed automatically, benefiting from masked finetuning.

3.6 Multimodal inpainting

Our method supports inpainting with a mixture of the aforementioned guidance. Mixed semantic guidance (text + exemplar) can be achieved by simply conditioning on the embedding c of combined exemplar token v^* and text w, i.e., $c = C(w, v^*)$. Mixed semantic-spatial guidance can be performed by additional stroke blending. Specifically, we begin by conditioning on null text $\emptyset$ in the early stages ($t > \tau$), where null text helps with unconditional completion at these steps. When $t = \tau$, we perform spatial stroke blending as described in Eq. 7, followed by semantic conditioning c for remaining steps ($t \leq \tau$). By adjusting τ, we can control the trade-off between realism and stroke-faithfulness. We show additional results in ablation study.

3.7 Masked attention control

While the model is finetuned to reconstruct the known region, we have observed an issue that the inpainted object may exceed the hole boundary and bleed into the known region, which may get truncated after blending, resulting in noticeable edge artifacts.

To mitigate this issue, we introduce masked attention control, the general idea is to restrict text attention with the known region in cross-attention layers in the diffusion model, and restrict the inpainted region's attention with the known region for self-attention layers, as illustrated in Fig. 3. Specifically, recall that in normal attention, query Q , key K and value V are mapped from image features for self-attention, while for cross-attention, K, V are mapped from textual features. Based on this, we introduce attention mask M^{attn}, and our masked attention is computed as follows:

$$\text{MaskedAttn}(Q, K, V) = \left[\text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right) \odot M^{attn}\right] \times V \quad (8)$$

where d is the dimension of K and V. Let n be the image feature size and l be the text feature length, and $\mathcal{I}$ be the set of known pixels' indices.

For cross-attention, M^{attn} has the size of $n^2 \times l$. Here, $M^{attn}[i, j]$ denotes whether the j^{th} textual token is allowed to attend to the i^{th} image pixel. To prevent text from affecting known region, we set $M^{attn}[i, j]$ as follows:

$$M^{attn}[i, j] = \begin{cases} 0, & \text{if } i \in \mathcal{I} \\ 1, & \text{if } i \notin \mathcal{I} \end{cases} \quad (9)$$

For self-attention, M^{attn} has the size of $n^2 \times n^2$. $M^{attn}[i, j]$ indicates whether the j^{th} image pixel is allowed to attend to the i^{th} image pixel. To prevent inpainted pixels from leaking into known region, we set $M^{attn}[i, j]$ as follows:

$$M^{attn}[i, j] = \begin{cases} 0, & \text{if } j \notin \mathcal{I} \text{ and } i \in \mathcal{I} \\ 1, & \text{otherwise} \end{cases} \quad (10)$$

We show in ablation studies that enabling masked attention control effectively prevents the generated object from overflowing outside the mask, thus avoiding truncation during blending.

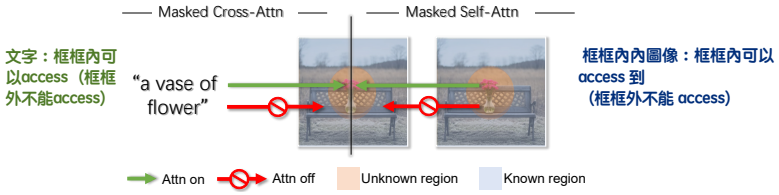

Figure 3: Illustration of masked attention control. For cross-attention (left), text can only attend to the unknown region but not the known region. For self-attention (right), inpainted content can only attend to the unknown region but not the known region.

4 EXPERIMENTS

4.1 Experiment setup

Implementation details. Our method is based on Stable Diffusion (sd-v1-4 checkpoint). We finetuned the model for 100 iterations using Adam optimizer [18] with default parameters, and a learning rate of 1e-5. We use the DDIM sampler [36] with 50 steps and a classifier-free scale of 8 [13].

Datasets. Our image samples were obtained from multiple sources, including: (1) EditBench [43], a systematic benchmark for text-driven image inpainting. We used its natural images and simple prompts as these are closer to practical use. (2) The Unsplash website. (3) Images captured by ourselves. (4) Images from other papers.

Evaluation metrics. We used the following popular quantitative metrics: (1) Neural Image Assessment (NIMA) [39]: a model-based reference-free perceptual image quality metric. (2) Text-to-image alignment (T2I) [11]: evaluates text-image CLIP similarity in text-driven inpainting. (3) Image-to-image alignment (I2I): evaluates the image-image CLIP similarity in exemplar-driven inpainting. (4) Root mean squared error (RMSE): assesses the faithfulness of the inpainted object to stroke color in stroke-driven inpainting. For these metrics, the test sample number varies from 50 to 120, depending on different tasks, and each sample's statistic is averaged from 8 diverse outputs. (5) Human preference: 55 participants were presented with 80 side-by-side result comparisons from different methods. To mitigate the choice bias, the order of the displayed options was randomized for each comparison. Participants were encouraged to vote for their most preferred result, but multiple selections were allowed (no more than half of the available options) if they found it hard to decide.

Baselines. We compared our methods with state-of-the-art diffusion-based inpainting methods. For unconditional and text-driven inpainting, we compared with RePaint [24] (unconditional only), SD-inpaint [32], GLIDE-inpaint [48], and Blended Latent Diffusion (BLD) [1]. For stroke-driven inpainting, we compared with SDEdit [25]. For exemplar-driven inpainting, we compared with Textual Inversion (TxtInv) [7] and Paint-by-Example [47].

4.2 Text-driven/unconditional inpainting

Qualitative comparison.We present a side-by-side visual comparison with related baselines in Fig. 4. As can be seen, our method is able to generate more plausible results than GLIDE-inpaint, BLD,

and RePaint. GLIDE-inpaint exhibits some artifacts along the hole boundary (as seen in the cheese cake example), while BLD tends to fit the mask shape, sometimes leading to unnatural transitions with the background (as seen with the over-sized leopard head). RePaint's repetitive sampling harmonizes the transitions but sometimes leads to incorrect inpainted semantics (see the bird with human head), since the image semantic is still not globally perceived by the model in essence. SD-inpaint and our method achieve comparable visual quality, but ours does not require training on a massive dataset.

Quantitative comparison.We report T2I (which reflects faithfulness to text), NIMA (which focuses on technical quality but is agnostic to aesthetics), and human votes (which reflect individual subjective preference, as a complement to NIMA) of different methods in Tab. 1. As can be seen, BLD obtains the highest T2I score but the lowest NIMA score, suggesting that it responds more strongly to the text, but sometimes this response is locally excessive and lead to unnatural oversized objects. Our method shows comparable NIMA with other methods, but differs in the image-specific optimization, thus was favored by most human evaluators. Moreover, we noticed that SD-Inpaint shows lesser text-image alignment compared to SD base model (as its T2I score is lower than ours). This is because SD-Inpaint was trained with randomly generated masks, which often cover image region unrelated to the text prompt. Training on such masked images encourages the model to ignore the text, resulting in a reduced or even absent textual responses, especially when the masked regions are small. This finding is also revealed by recent studies [43, 45]. Therefore, our work builds on SD base model for its superior textual capability, which also brings more potential to other modalities.

Table 1: Quantitative comparison on text-driven and unconditional (in parentheses) inpainting.

	T2I	NIMA	Human votes
RePaint [24]	-	(4.48)	(26.73%)
BLD [1]	**26.71**	4.25 (4.21)	29.64% (15.73%)
SD-Inpaint [32]	25.68	**4.63** (**4.53**)	32.82% (36.73%)
GLIDE-Inpaint [48]	22.95	4.51 (4.45)	27.09% (21.00%)
Ours	26.48	4.59 (4.49)	**38.64**% (**56.82**%)

4.3 Exemplar-driven inpainting

Qualitative comparison. We compared our results with Paint-by-Example [47], and the combined implementation of TxtInv and BLD as done in [7], denoted as TxtInv+BLD. Our subject tokens were automatically determined using Eq. 6. The results are shown in Fig. 5, where we can see our method can better retain details from the exemplar. Paint-by-Example achieves plausible results for commonly-seen concepts (e.g., cat and dog) but falls short when presented with less common or customized objects that may not appear in their training dataset. TxtInv+BLD inherits the similar truncation issues from BLD, and produces less aligned results than ours. We attribute this to the fact that fine-tuning the model normally has a stronger fitting ability compared to solely optimizing the word embedding.

Quantitative comparison. We use I2I score to measure the image similarly between the inpainted part and the exemplar image. We also report NIMA score and human preference in Tab. 2. While all these methods generate images of comparable quality, ours excels at capturing the semantics of the exemplar, particularly for personalized concepts, resulting in higher scores for both the I2I metric and human votes.

Table 2: Quantitative results on exemplar-driven inpainting.

	I2I	NIMA	Human votes
TxtInv.+BLD [1, 7]	78.24	5.25	9.36%
Paint-by-Ex. [47]	77.75	**5.32**	21.27%
Ours	**78.41**	5.28	**69.36**%

4.4 Stroke-driven inpainting

Qualitative comparison. Since there is no stroke-driven inpainting baseline so far, we compared our method with the combined implementation of SDEdit [25] and BLD [1], denoted as SDEdit+BLD. As shown in Fig. 6, SDEdit+BLD succeeds in generating objects that are well-aligned with the strokes. However, it fails to fill the remaining unknown area with plausible content where stroke hints are absent (see the apples in Fig. 6). In contrast, our approach accomplishes both stroke faithfulness and background completion, enabling users to focus on their interested objects without having to scribble over the entire missing area. This reveals that the proposed masked finetuning helps the model gain awareness of image semantics, bringing plausible completion in the unknown region.

Quantitative comparison. Fig. 7 presents quantitative statistics on two aspects: color faithfulness to stroke, measured by 1-RMSE (flipped RMSE), and image quality, measured by NIMA score. There exits a trade-off between them when choosing different stroke blending timestep τ. As can be seen, at the same level of stroke faithfulness, our results generally have a better quality. We attribute this to better background preservation and higher degree of editing in our approach. For human evaluation, our method received 54.64% of the votes as opposed to 45.36% of SDEdit+BLD.

拿1去减

4.5 Inpainting with mixed guidance

Uni-paint stands out from previous approaches by supporting the use of mixed multimodal guidance for inpainting task. We demonstrate this capability in Fig. 8, where text or exemplar are used to deliver the subject's semantic attributes and stroke is used to determine its color and layout. By sampling from different initial noise, our method can generate diverse outputs given the same input and guidance.

4.6 Ablation studies

We conduct ablation studies on several settings used in our work.

Masked finetuning. To demonstrate the benefits of masked finetuning, we present visual examples with different finetuning iterations in Fig. 9. Without masked finetuning (0 iters), the model only focuses on the local region and disregards the context, resulting in noticeable stitching artifacts. This issue is mitigated after 75

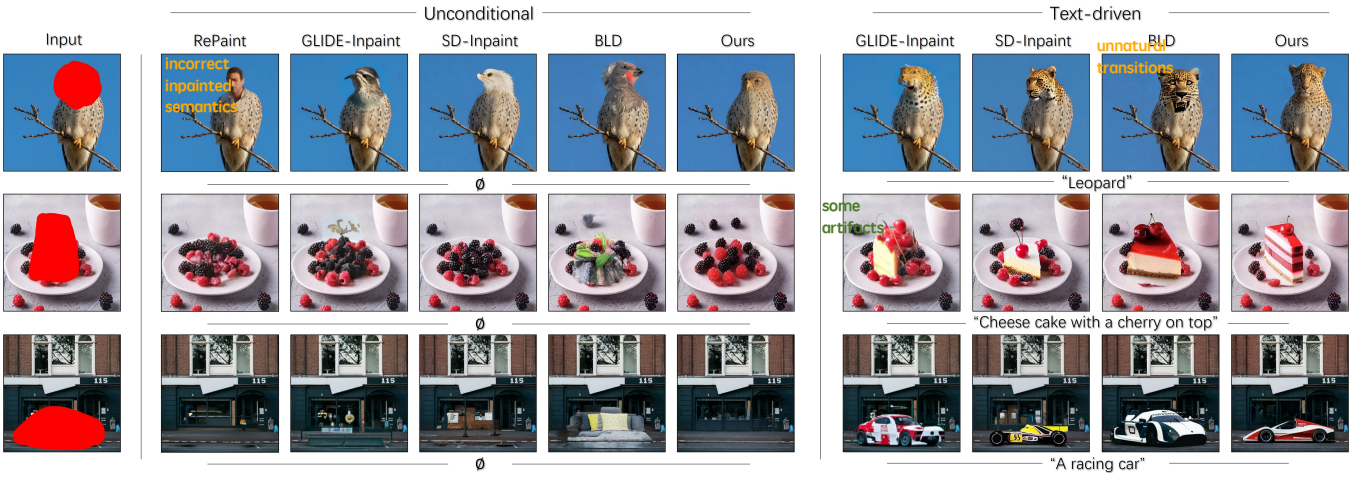

Figure 4: Qualitative comparison on unconditional (col.2-6) and text-driven (col.7-10) inpainting with related methods.

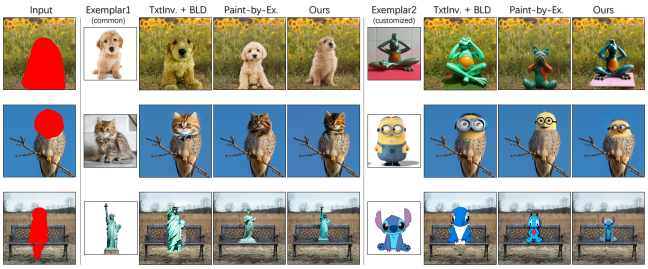

Figure 5: Qualitative comparison on exemplar-driven inpainting with related methods.

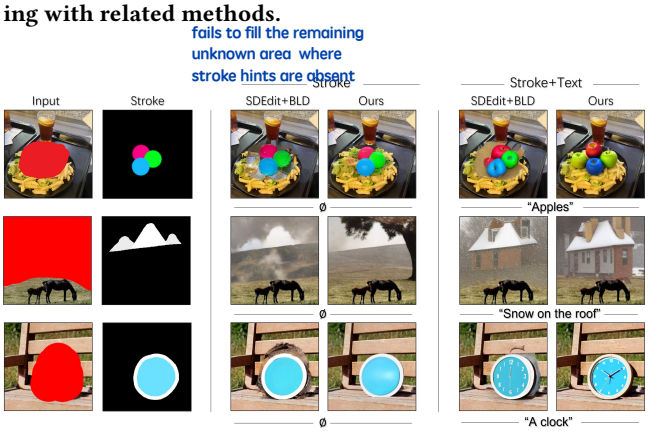

Figure 6: Qualitative comparison on stroke-driven inpainting.

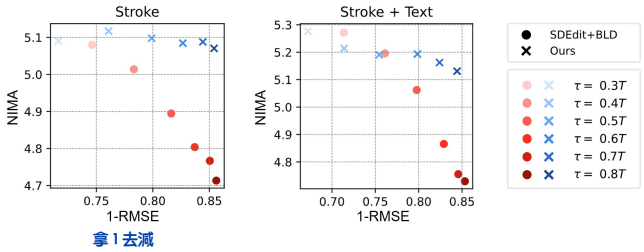

Figure 7: Quantitative comparison of inpainting quality (NIMA score) and stroke-alignment (1-RMSE) on stroke-driven (left) and stroke+text-driven (right) inpainting with different τ. Upper right corner indicates a better trade-off.

iterations of finetuning. This suggests that masked finetuning helps the model gain semantic awareness in the known region, leading to coherent texture transitions in inpainted region.

Masked attention control. We introduce masked attention control mainly to suppress over-sized inpainted content from leaking into the known area. To demonstrate this effect, we generate images of tigers with and without applying masked attention in two scenarios: free-generation and inpainting, as shown in Fig. 10. In the free-generation case, enabling masked attention control effectively constrains the generation scope within the masked region. In the inpainting case, disabling masked attention control can sometimes result in over-sized inpainted content, which may get truncated after background blending. This suggests that restricting the attention flow can be useful in local editing tasks like inpainting.

Stroke blending timestep. In stroke-driven inpainting, threshold τ adjusts the balance between realism and stroke-faithfulness, we show a series of visual results with different choices of τ in Fig. 11, quantitative statistics can be found in Fig. 7. Generally, larger τ leads to more realistic but less aligned results, $\tau \in [0.5T, 0.6T]$ normally yields a balanced effect (T is the total number of timestep).

5 CONCLUSION

We propose Uni-paint, a unified multimodal inpainting framework supporting unconditional, text, stroke, and exemplar guidance. Our unconditional and text-driven inpainting results are competitive with recent works without large-scale training. For exemplar-driven inpainting, our few-shot approach achieves improved customization effects. Stroke guidance on regions of interest is also integrated

Figure 8: Diverse inpainting results under mixed guidance from a combination of text, stroke, and exemplar.

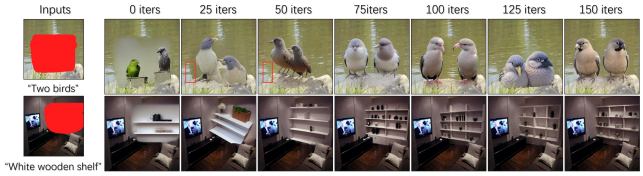

Figure 9: Effect of masked finetuning with different iterations. Applying masked finetuning enhances model's context awareness and brings better texture transition.

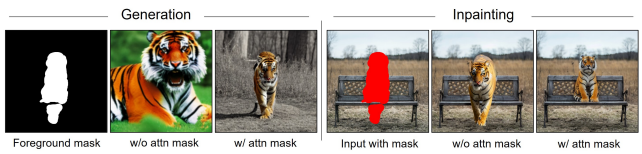

Figure 10: Examples of generation/inpainting of tiger with/without masked attention control. Applying masked attention control effectively constrains generation scope.

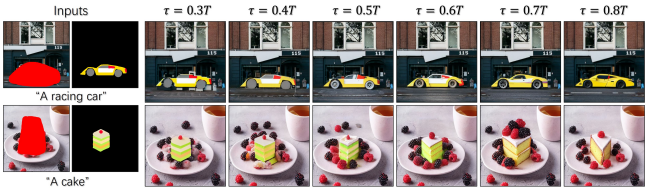

Figure 11: Effect of different τ on stroke-driven inpainting. Larger τ leads to more realistic but less aligned results.

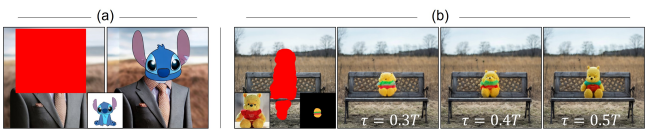

Figure 12: Failure cases: (a) Unnatural stitching. (b) Failed to obey conflicting guidance (stroke shape and exemplar)

in our framework. Moreover, our method supports inpainting with mixed guidance, which is not available in existing methods.

However, our method still encounters some limitations. First, when there is a large gap between the exemplar and the input (e.g., cartoon vs. real images), our method may fail to fully harmonize the gap, resulting in unnatural stitching (see Fig.12a). Second, conflicts may occur when mixing guidance from different modalities. For example, in Fig.12b, the hamburger-shaped stroke is far different from the exemplar, making it challenging to find an appropriate τ that simultaneously respects both the stroke and the exemplar guidance. Note this issue can be avoided with careful interactions.

Our future work aims to address these limitations and explore additional modalities, further investigating the potential of diffusion models in image inpainting.

ACKNOWLEDGMENTS

This work is supported by GRF grant from the Research Grants Council (RGC) of Hong Kong. We also thank Unsplash and the photographers for generously sharing their high-quality, free-to-use images used in this research.

REFERENCES

[1] Omri Avrahami, Ohad Fried, and Dani Lischinski. 2023. Blended latent diffusion. *ACM Transactions on Graphics (TOG)* 42, 4 (2023), 1–11.
[2] Omri Avrahami, Dani Lischinski, and Ohad Fried. 2022. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18208–18218.
[3] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. 2009. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.* 28, 3 (2009), 24.
[4] Tim Brooks, Aleksander Holynski, and Alexei A Efros. 2023. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18392–18402.
[5] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. 2023. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics (TOG)* 42, 4 (2023), 1–10.
[6] Antonio Criminisi, Patrick Pérez, and Kentaro Toyama. 2004. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on image processing* 13, 9 (2004), 1200–1212.
[7] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. 2022. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618* (2022).
[8] Ligong Han, Yinxiao Li, Han Zhang, Peyman Milanfar, Dimitris Metaxas, and Feng Yang. 2023. SVDiff: Compact Parameter Space for Diffusion Fine-Tuning. *arXiv preprint arXiv:2303.11305* (2023).
[9] Kaiming He and Jian Sun. 2014. Image completion approaches using the statistics of similar patches. *IEEE transactions on pattern analysis and machine intelligence*

36, 12 (2014), 2423–2435.
[10] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. 2022. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626* (2022).
[11] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718* (2021).
[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems* 33 (2020), 6840–6851.
[13] Jonathan Ho and Tim Salimans. 2022. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598* (2022).
[14] Dong Huk Park, Grace Luo, Clayton Toste, Samaneh Azadi, Xihui Liu, Maka Karalashvili, Anna Rohrbach, and Trevor Darrell. 2022. Shape-Guided Diffusion with Inside-Outside Attention. *arXiv e-prints* (2022), arXiv–2212.
[15] Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4401–4410.
[16] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. 2023. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6007–6017.
[17] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. 2022. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2426–2435.
[18] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
[19] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. 2023. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1931–1941.
[20] Jingyuan Li, Fengxiang He, Lefei Zhang, Bo Du, and Dacheng Tao. 2019. Progressive reconstruction of visual structure for image inpainting. In *Proceedings of the IEEE/CVF international conference on computer vision*. 5962–5971.
[21] Jingyuan Li, Ning Wang, Lefei Zhang, Bo Du, and Dacheng Tao. 2020. Recurrent feature reasoning for image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7760–7768.
[22] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. 2018. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European conference on computer vision (ECCV)*. 85–100.
[23] Xihui Liu, Dong Huk Park, Samaneh Azadi, Gong Zhang, Arman Chopikyan, Yuxiao Hu, Humphrey Shi, Anna Rohrbach, and Trevor Darrell. 2023. More control for free! image synthesis with semantic diffusion guidance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 289–299.
[24] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. 2022. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11461–11471.
[25] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. 2021. Sdedit: Image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073* (2021).
[26] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. 2023. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6038–6047.
[27] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Z Qureshi, and Mehran Ebrahimi. 2019. Edgeconnect: Generative image inpainting with adversarial edge learning. *arXiv preprint arXiv:1901.00212* (2019).
[28] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. 2021. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2085–2094.
[29] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. 2016. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2536–2544.
[30] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PMLR, 8748–8763.
[31] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125* (2022).
[32] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10684–10695.
[33] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. 2023. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 22500–22510.
[34] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. 2022. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*. 1–10.
[35] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems* 35 (2022), 36479–36494.
[36] Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502* (2020).
[37] Yuhang Song, Chao Yang, Zhe Lin, Xiaofeng Liu, Qin Huang, Hao Li, and C-C Jay Kuo. 2018. Contextual-based image inpainting: Infer, match, and translate. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 3–19.
[38] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. 2022. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*. 2149–2159.
[39] Hossein Talebi and Peyman Milanfar. 2018. NIMA: Neural image assessment. *IEEE transactions on image processing* 27, 8 (2018), 3998–4011.
[40] Dani Valevski, Matan Kalman, Yossi Matias, and Yaniv Leviathan. 2022. Unitune: Text-driven image editing by fine tuning an image generation model on a single image. *arXiv preprint arXiv:2210.09477* (2022).
[41] Andrey Voynov, Kfir Aberman, and Daniel Cohen-Or. 2022. Sketch-Guided Text-to-Image Diffusion Models. *arXiv preprint arXiv:2211.13752* (2022).
[42] Ziyu Wan, Jingbo Zhang, Dongdong Chen, and Jing Liao. 2021. High-fidelity pluralistic image completion with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4692–4701.
[43] Su Wang, Chitwan Saharia, Ceslee Montgomery, Jordi Pont-Tuset, Shai Noy, Stefano Pellegrini, Yasumasa Onoe, Sarah Laszlo, David J Fleet, Radu Soricut, et al. 2023. Imagen editor and editbench: Advancing and evaluating text-guided image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18359–18369.
[44] Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. 2023. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. *arXiv preprint arXiv:2302.13848* (2023).
[45] Shaoan Xie, Zhifei Zhang, Zhe Lin, Tobias Hinz, and Kun Zhang. 2023. Smartbrush: Text and shape guided object inpainting with diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 22428–22437.
[46] Wei Xiong, Jiahui Yu, Zhe Lin, Jimei Yang, Xin Lu, Connelly Barnes, and Jiebo Luo. 2019. Foreground-aware image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5840–5848.
[47] Binxin Yang, Shuyang Gu, Bo Zhang, Ting Zhang, Xuejin Chen, Xiaoyan Sun, Dong Chen, and Fang Wen. 2023. Paint by example: Exemplar-based image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18381–18391.
[48] Han Yang, Ruimao Zhang, Xiaobao Guo, Wei Liu, Wangmeng Zuo, and Ping Luo. 2020. Towards photo-realistic virtual try-on by adaptively generating-preserving image content. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 7850–7859.
[49] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. 2018. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5505–5514.
[50] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. 2019. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE/CVF international conference on computer vision*. 4471–4480.
[51] Haoran Zhang, Zhenzhen Hu, Changzhi Luo, Wangmeng Zuo, and Meng Wang. 2018. Semantic image inpainting with progressive generative networks. In *Proceedings of the 26th ACM international conference on Multimedia*. 1939–1947.

Appendix

In this appendix, we provide additional details and results of our approach. We present more implementation details in Sec. A. Additional human evaluation details are given in Sec.B. We also provide more visual result of unconditional, text-driven, stroke-driven, exemplar-driven, and multimodality inpainting in Sec. C.

A IMPLEMENTATION DETAILS

A.1 Masked finetuning

We introduced our general motivation and core steps of masked finetuning in Sec. 3.2 of the main paper. Here we present more detailed process in pseudo code format in Algorithm 1.

Algorithm 1 Masked finetuning

Require: Input image X^{in}, binary mask M, exemplar X^{ref} (optional), pretrained stable diffusion model ϵ_θ, text encoder C, image encoder E.

Get image latent $x^{in} = E(X^{in} \odot M)$
Get latent mask $m = \text{Resize}(M)$ s.t. m has the same size as x^{in}
Get null text embedding $\emptyset = C(\text{""})$
Get exemplar token v^* from Eq. 6 if X^{ref} exists
while $iter < total\_iters$ **do**
  $t_1 \sim \mathcal{U}(0, T)$
  $\epsilon_1 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  $x^{in}_{t_1} = \sqrt{\alpha_{t_1}} x^{in} + \sqrt{1-\alpha_{t_1}}\epsilon_1$
  $\ell_{bg} = \left\| m \odot \epsilon_1 - m \odot \epsilon_\theta \left(x^{in}_{t_1}, \emptyset, t_1 \right) \right\|_2^2$
  if exist(X^{ref}) **then**
    $t_2 \sim \mathcal{U}(0, T)$
    $X^{ref}, M^{ref} = \text{RandomShiftAndScale}(X^{ref})$
    $x^{ref} = E(X^{ref})$
    $m^{ref} = \text{Resize}(M^{ref})$
    $\epsilon_2 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
    $x^{ref}_{t_2} = \sqrt{\alpha_{t_2}} x^{ref} + \sqrt{1-\alpha_{t_2}}\epsilon_2$
    $\ell_{ref} = \left\| m^{ref} \odot \epsilon_2 - m^{ref} \odot \epsilon_\theta \left(x^{ref}_{t_2}, C(v^*), t_2 \right) \right\|_2^2$
  else
    $\ell_{ref} = 0$
  end if
  $\ell = \ell_{bg} + \ell_{ref}$
  $\theta = \theta - lr \cdot \nabla_\theta \ell$
end while
Return Finetuned model ϵ_{θ^*}

A.2 Auto-subject token visualization

In Sec. 3.4, we developed a CLIP-based method for automatically obtaining the initial subject token v^* from a given exemplar, which can be useful in cases where the user is uncertain about the subject category or in automated application scenarios. Note that before finetuning the model, v^* only provides a rough initial approximation but not perfect alignment to the exemplar. We present some visualization examples of this approach in Fig. 13. In these examples, the bottom-row images are generated conditioned on v^* by an **un-tuned** model obtained from the top-row exemplar images. We found that for well-known concepts such as the Statue of Liberty, v^* is directly capable of reproducing the exemplar. For less-common concepts (e.g., cartoon characters Stitch and Minion), while the details are not perfectly preserved, v^* still provides a rough initial approximation for capturing the exemplar concepts. After the model is fine-tuned, v^* will be bound with exemplar and is able to produce aligned results.

A.3 Computational speed

With a batch size of 1, our model takes roughly 98 seconds for finetuning with 100 iterations, and 4.8 seconds for inference with 50 DDIM steps on a NVIDIA A6000 GPU, which is the typical speed of official released stable diffusion model without using acceleration strategy or model compression. Note that the speed can be further accelerated by using more advanced sampler (e.g., DPM-Solver++) or toolbox (e.g., xFormers).

B HUMAN EVALUATION DETAILS

B.1 Setup

As described in Sec. 4.1 in the main paper, we conducted a user study in questionnaire format to determine which method produces the best results in terms of human perception. We invited a total of 55 participants. None of the participants were involved in this research in part or in whole, or had any conflicts of interest. The questionnaire consists of 4 sections for 4 different guidance: unconditional inpainting, text-driven inpainting, exemplar-driven inpainting, and stroke-driven inpainting, respectively. Each section has 20 side-by-side comparisons of different methods. To mitigate the potential choice bias, the displayed order of options was randomly shuffled for each question and each participant. For each question, participants were encouraged to choose their most preferred option, but in case they found it hard to decide, they were allowed to make multiple choices but no more than half of the available options (i.e., up to 2 options for unconditional and text-driven tasks, and 1 option only for exemplar-driven and stroke-driven tasks). Fig. 14 shows some example questions from our questionnaire.

B.2 Statistics

We also show detailed human evaluation statistics from two aspects: votes percentage per question (refers to the percentage of votes received by a method, relative to the total number of votes for a question), and votes percentage per participant (refers to the percentage of votes received by a method, relative to the total number of votes cast by a participant), as demonstrated in Fig. 15. As can be seen, our unconditional and text-driven inpainting results are roughly comparable to SD-Inpaint, with a slight numerical advantage. Our exemplar-driven and stroke-driven results received more votes in a greater number of questions and were favored by more of voters.

C ADDITIONAL VISUAL RESULTS

We provide additional visual comparison with the same baseline methods as in the main paper. We show more unconditional, text-driven, exemplar-driven and stroke-driven result comparison in

Figure 13: Visualization examples of automatic subject token identification. The bottom-row images are generated by un-tuned model conditioned on subject token v^* obtained from the top-row exemplar images, suggesting that v^* is able to provide a rough initial approximation to the exemplar.

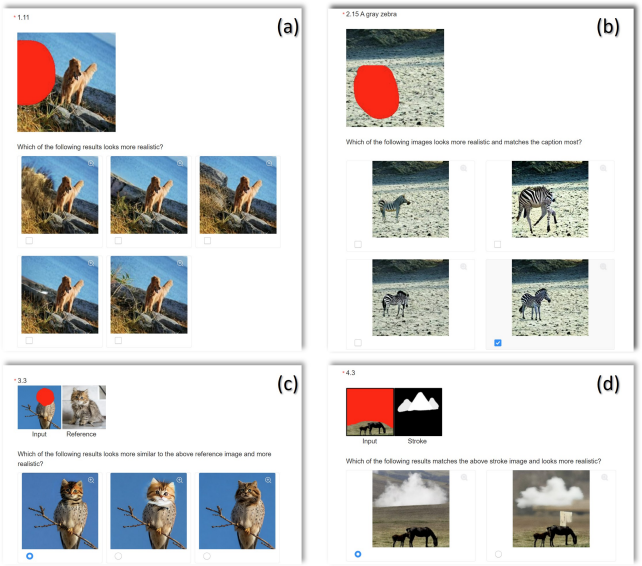

Figure 14: Interface of human evaluation questionnaire, these are four example questions of four different inpainting task: (a) Unconditional inpainting, (b) Text-driven inpainting, (c) Exemplar-driven inpainting, (d) Stroke-driven inpainting. Participants can select up to 2 options for (a) and (b), and 1 option only for (c) and (d).

Fig. 16, 17, 18, and 19, respectively. We also present additional results of mixed guidance in Fig. 20.

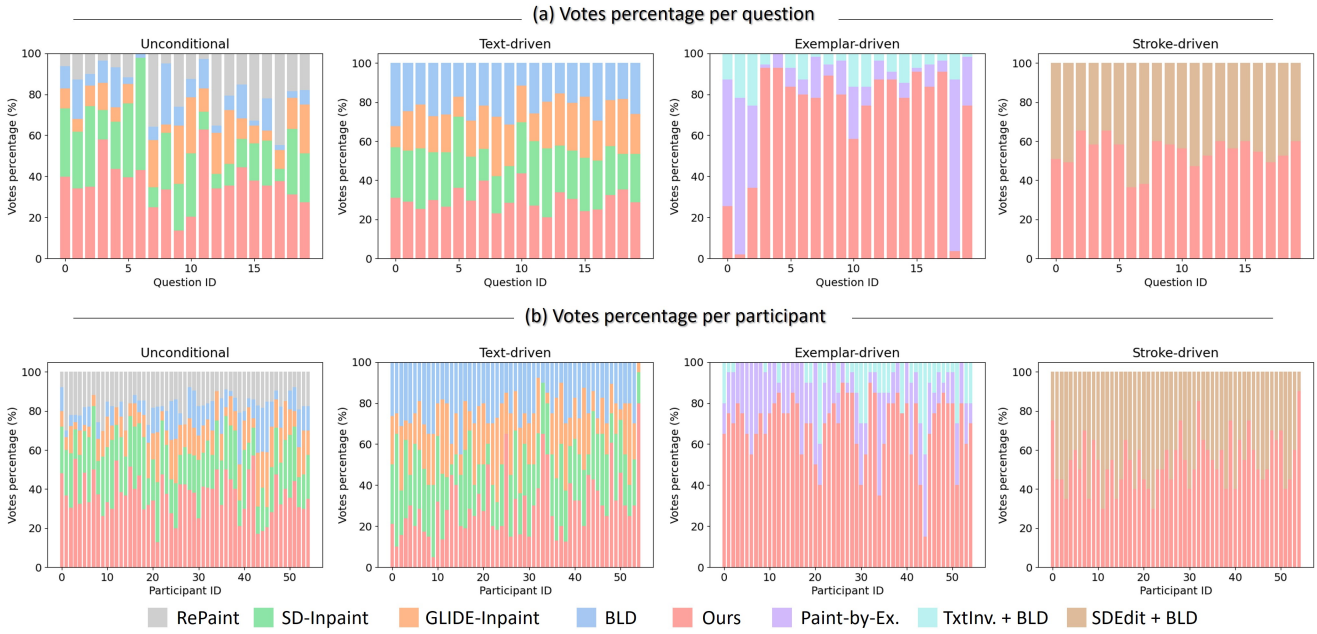

Figure 15: Detailed human evaluation statistics. (a) presents the votes percentage of each question. (b) presents the votes percentage of each participant.

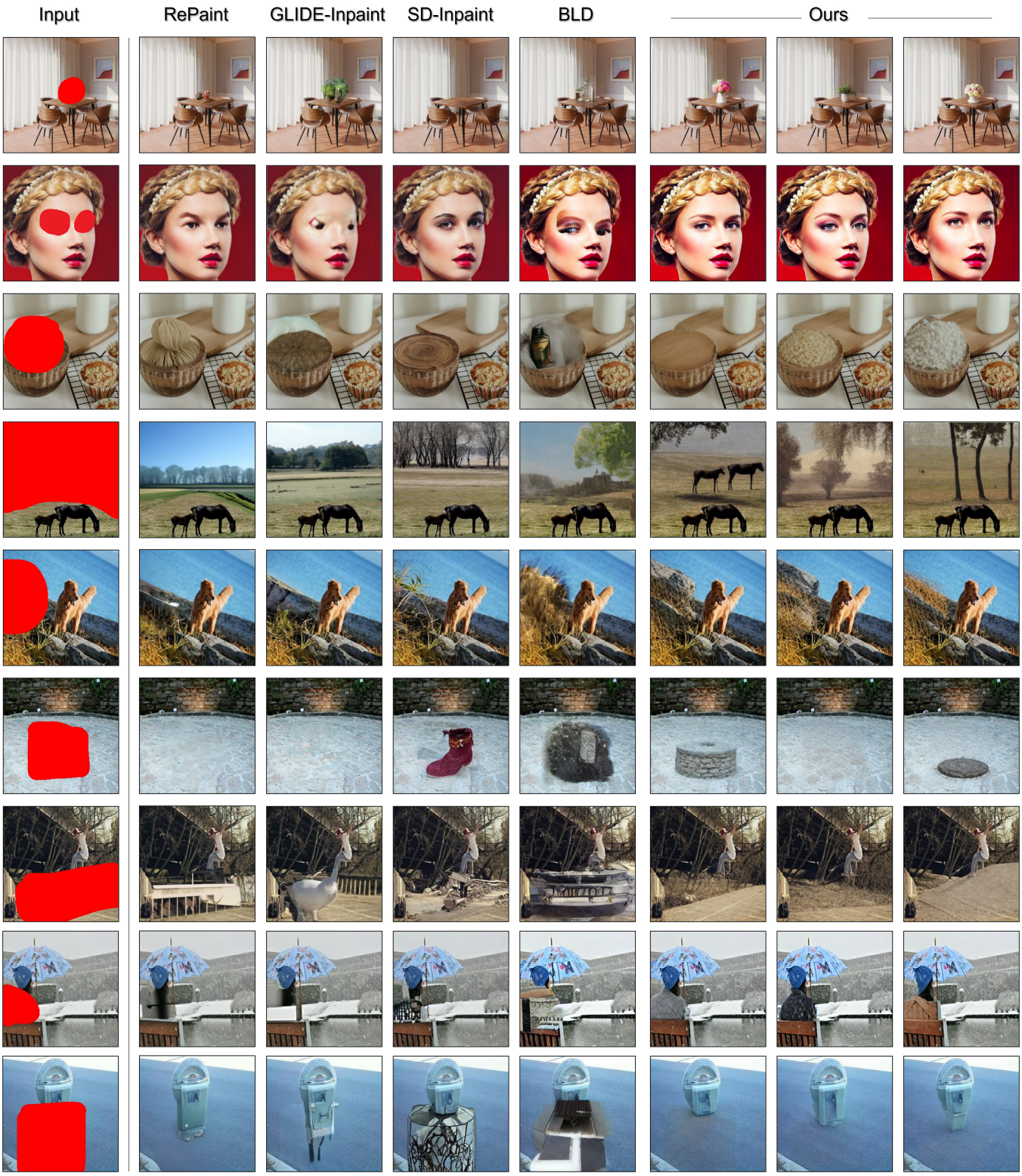

Figure 16: Additional unconditional inpainting results.

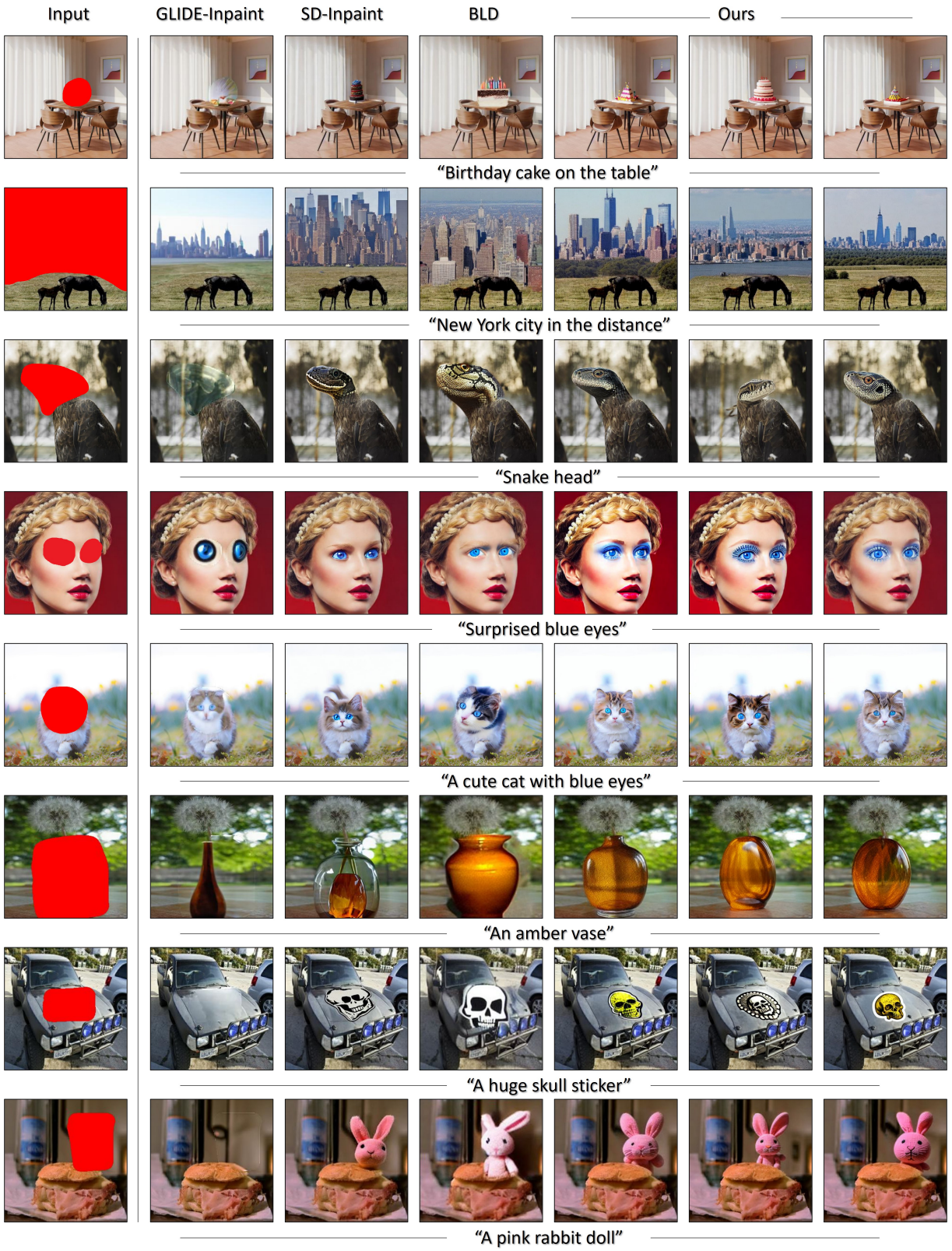

Figure 17: Additional text-driven inpainting results.

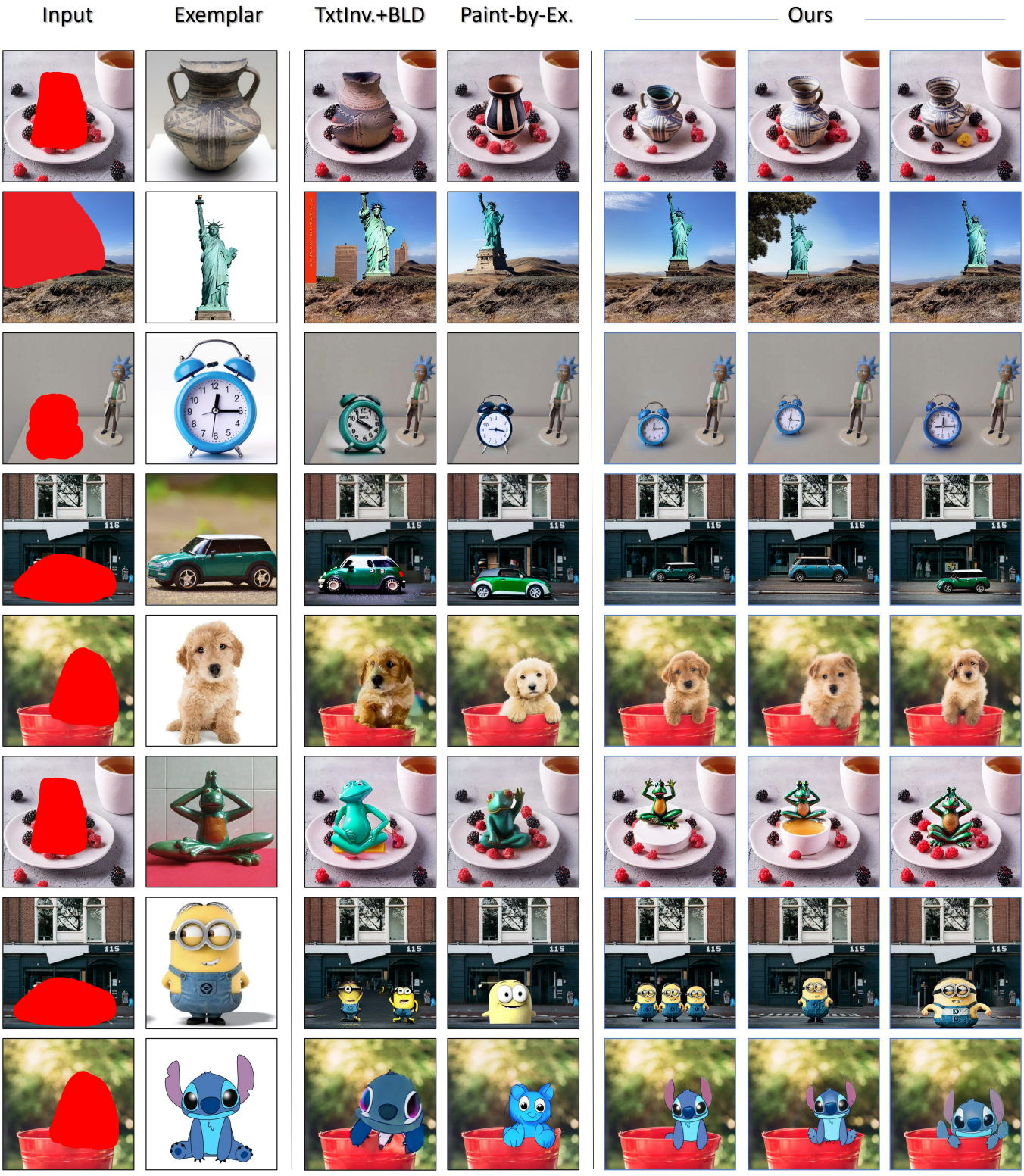

Figure 18: Additional exemplar-driven inpainting results.

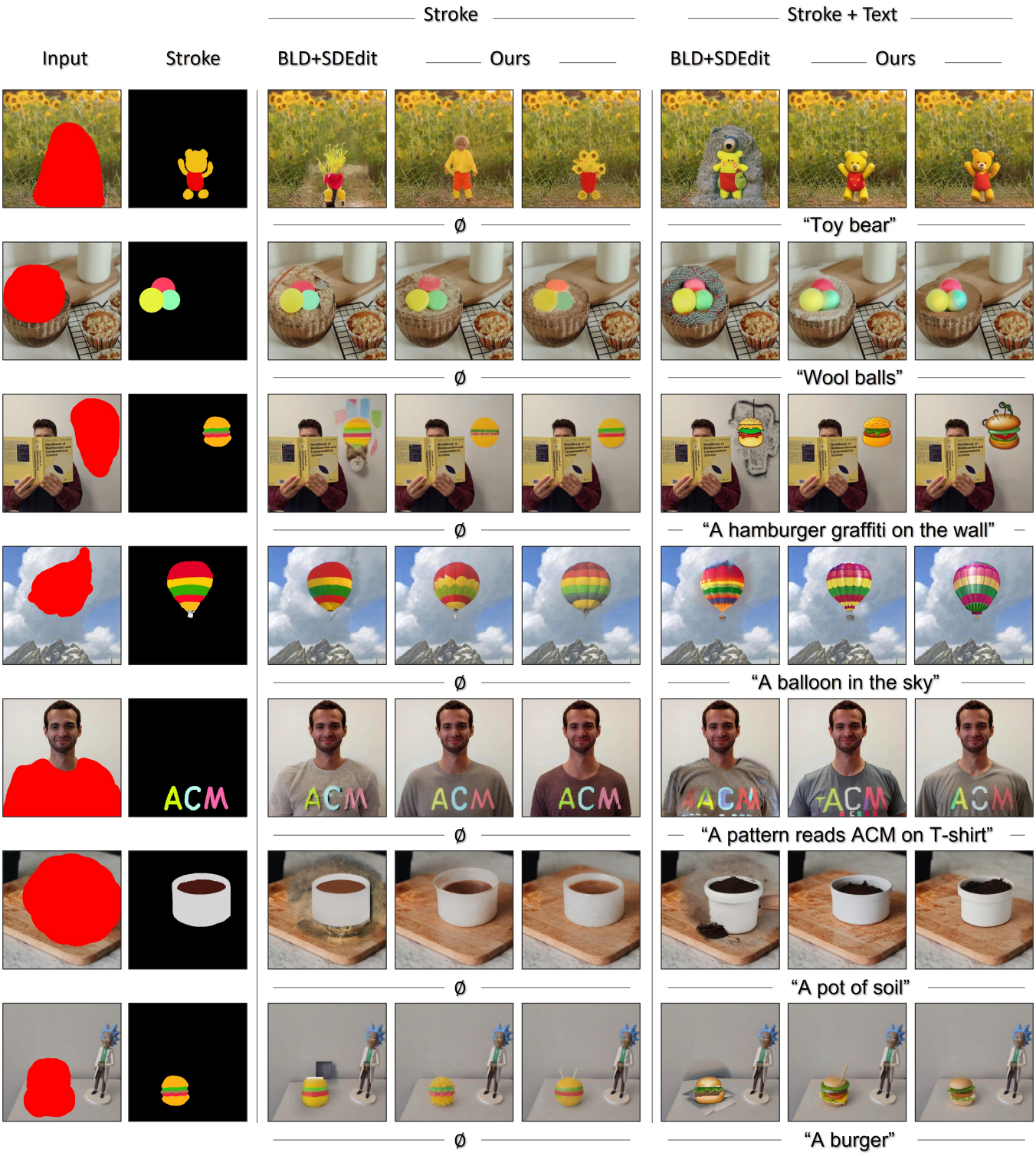

Figure 19: Additional stroke-driven inpainting results.

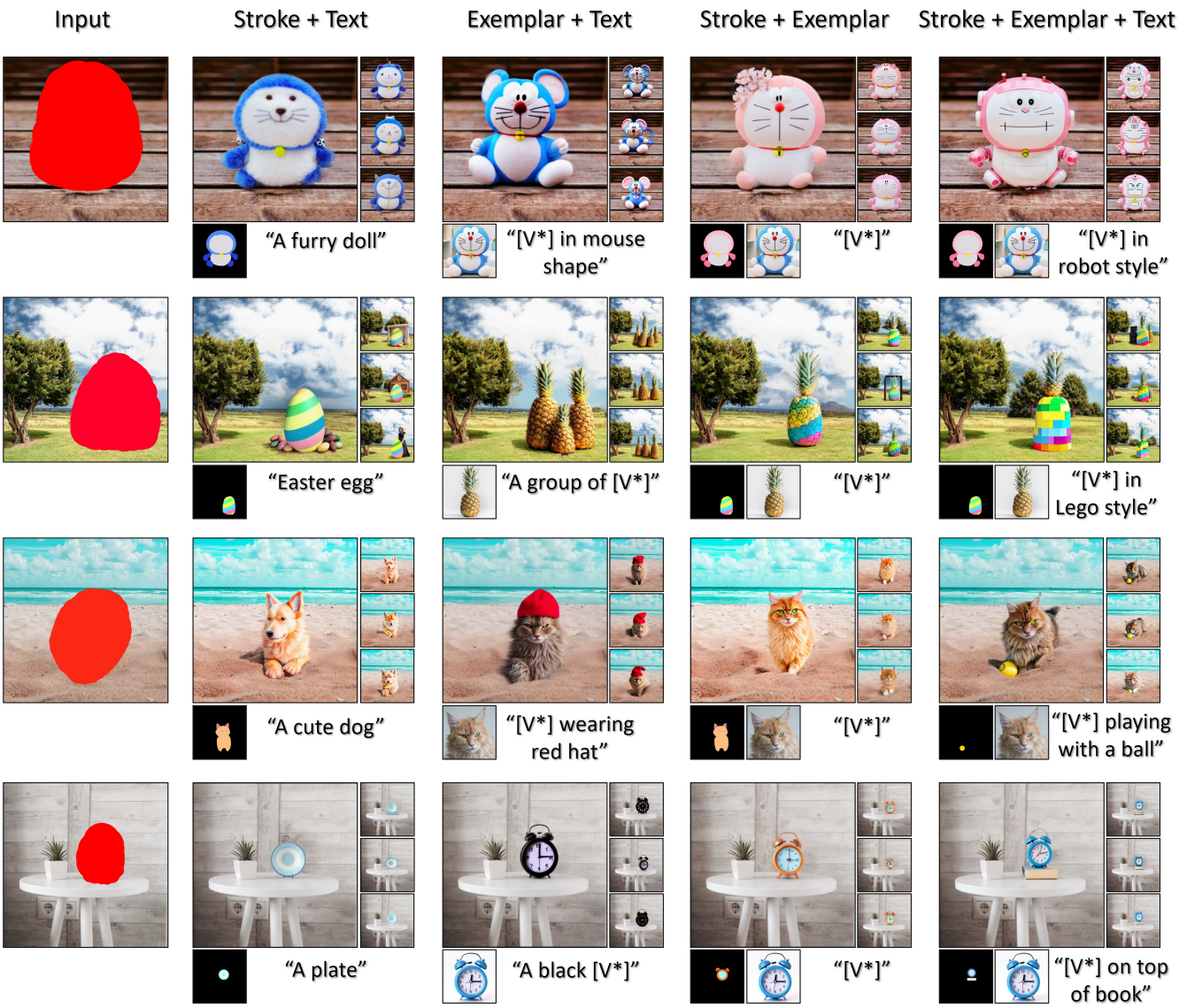

Figure 20: Additional inpainting results of mixed-guidance.