UNet-Twice: A Simple Structured Reference-based Inpainting Framework

Junda Lu[1], Zhiqiao Xu[2], Houkun Wu[2], Wei Cui[3], Bo Huang[4,5], Mingyang Chen[5], and Bing Li[2](✉)

[1] Western Sydney University, Australia
junda.lu@westernsydney.edu.au
[2] University of Electronic Science and Technology of China, China
{zhiqiao_xu, houkun_wu}@std.uestc.edu.cn, bing_li@uestc.edu.cn
[3] Institute for Infocomm Research (I^{2}R), Agency for Science, Technology and Research (A*STAR), Singapore
cui_wei@a-star.edu.sg
[4] The Hong Kong University of Science and Technology (Guangzhou), China
[5] The Hong Kong University of Science and Technology, Hong Kong SAR, China
{bhuangas, mchenbt}@connect.ust.hk

Abstract. Reference-based inpainting utilizes the information in the reference image to inpaint the missing region in the target image, aiming to generate the fact object behind the mask rather than a reasonable one. The key challenge is how to accurately locate the region in the reference image and maximally utilize it. We proposed UNet-Twice, a simple yet novel framework. UNet-Twice uses the same UNet two times, combined with a well-constructed data pair to build a close connection between the target and reference image. The two passes fuse spatially complementary features generated from the target and the reference image, which provides intuitively explicit guidance for image generation. Our proposed framework is simple and easy to extend to a multi-reference setting. Extensive experiments demonstrate that our proposed framework outperforms existing reference-based inpainting methods in both single-reference and multi-reference settings.

Keywords: Reference-based Inpainting · Image Inpainting · Diffusion Model

1 Introduction

Image inpainting is a fundamental task in computer vision that aims to restore missing regions in an image using surrounding visual context [22]. Recent advances in Text-to-Image diffusion models [3,8,15,26,30,34,36,42] have demonstrated remarkable generative capability, making diffusion backbones widely adopted for image inpainting [27]. These models synthesize highly realistic content and produce visually plausible results even under large missing regions. However, diffusion models are better at generating visually realistic content than

preserving factual consistency, often producing plausible results that fail to faithfully recover scene-specific content. Reference-based inpainting [2,7,44,45,51,61] aims to address this limitation by introducing an additional reference image that provides appearance guidance for the missing region (e.g., texture, material, or object-specific details). Unlike generic inpainting, the goal is not merely to generate plausible content, but use the additional image to restore the correct content details from the reference image.

Despite the availability of an additional reference image, effectively exploiting reference information remains non-trivial. The core difficulty lies in maximizing the utility of the reference image, which involves two coupled sub-problems: *i*) *correspondence localization*, identifying regions in the reference that correspond to the masked region in the target. This is complicated by differences in viewpoint, scale, and layout that hinder reliable spatial estimation. *ii*) *content extraction and integration*, transferring relevant appearance patterns while maintaining global consistency. Simple copying often disrupts target image's underlying semantic and structural consistency.

To tackle these challenges, existing efforts explore several strategies, yet each has limitations. Straightforward solutions [44, 51] directly fine-tune diffusion models using reference images as additional training data. However, the absence of explicit spatial connections between the target and reference images limit faithful content transfer. Geometry-alignment-based methods [24,60,61] improve reference utilization through geometric alignment to provide spatial guidance. However, their effectiveness depends on the accuracy of geometric estimation. More recent architecture-augmented methods [6, 17, 45, 56] introduce additional model branches (e.g., a second UNet [35]) to extract reference features. While remarkably effective, such designs often come at the cost of increased architectural complexity, making the models harder to deploy and maintain.

Motivated by this observation, we seek a simple yet elegant alternative. We propose UNet-Twice, a reference-guided diffusion framework that reuses the *same* UNet backbone with two passes within a single forward process. As shown in Fig. 1, in the first pass, the UNet extracts locally reference-aware latent representations from the reference image. In the second pass, the masked target image is processed, and the reference features extracted in the first pass are injected into the denoising process to enable reference-conditioned synthesis. Since both passes are produced by the same UNet weights, these features lie in a compatible feature space and can be injected by simple element-wise addition, without any additional bridging module.

Building on this weight-sharing design, we further introduce a paired input formulation with identical input structures to avoid ambiguity between the two passes, which explicitly establishes an image-level association between the reference and target images. Furthermore, the reference image is geometrically aligned with the target image and modulated using a complementary mask, making it spatially complementary to the visible target region. Such complementary conditioning introduces a region-level connection, allowing the network to fuse "what is missing" in the target image with "what the reference suggests" through the

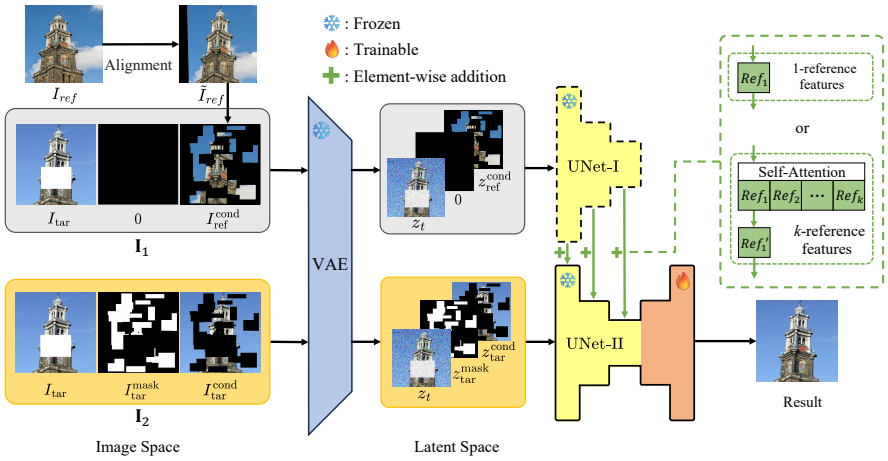

Fig. 1: UNet-Twice Pipeline. UNet-Twice reuses the same UNet twice within a single forward process. In the first pass, UNet-I extracts reference features from the reference image set $\mathbf{I}_1$. In the second pass, UNet-II takes the target image set $\mathbf{I}_2$ as the standard inpainting input and performs denoising. Since both passes share identical parameters, the extracted features lie in the same feature space and can be injected into the corresponding layers of UNet-II via element-wise addition, without introducing additional branches or bridging modules. For k references, UNet-I is repeated k times to extract k reference features, which are then aggregated via self-attention into a single feature for injection.

strong representation and feature fusion capability of the UNet. Importantly, due to the robustness of latent diffusion representations, perfect alignment is not required. Instead, it only requires the transformed reference content to approximately fall within the corresponding region, which substantially relaxes the requirement on correspondence estimation.

Our contributions are summarized as follows:

- We propose UNet-Twice, a reference-based inpainting framework with a two-pass design that reuses the *same* diffusion UNet twice within a single forward process. It integrates reference features into the image synthesis process without a separate reference branch or bridging module.
- We introduce a complementary conditioning formulation that, together with the two-pass design, explicitly couples the target and reference images at both the image and region levels. Through paired inputs and spatially complementary masking, the framework enables effective reference feature extraction and faithful content transfer.
- Extensive experiments validate the effectiveness and efficiency of the proposed framework across both single-reference and multi-reference settings. Detailed ablation studies validating each component of the design.

2 Related Work

2.1 Image Inpainting

Image inpainting is a fundamental task that fills the masked region with consistent context. Traditional methods [1, 9, 55] focus on low-level patterns thus results in limited performance. Subsequent works achieves promising results by using transformer-based [11, 18, 19, 25, 40] or GAN-based [5, 10, 14, 23, 38, 46] inpainting with adaptive convolutions [22,53], and attention [4,18,52,54]. In recent years, methods [33,34,37,43,49,50,59] based on the diffusion model demonstrated high image inpainting capabilities. diffusion-based image inpainting architecture design. ASUKA [47] inpaint the image by introducing additional constraints to preserve structural consistency and semantic coherence during generation. LOHC [58] adopts a cooperative reconstruction strategy that jointly predicts structural priors and image appearance for large occlusion completion. BrushNet [17] proposes a plug-and-play dual-branch diffusion framework that explicitly separates masked image features from noisy latents to enhance content and shape-aware image inpainting.

2.2 Reference-based Image Inpainting

Reference-based image inpainting has attracted increasing attention in recent years, aiming to improve image completion by leveraging external reference images to provide additional appearance and semantic cues. Despite differences in architectural design and conditioning strategies, existing methods share a common objective: to maximally exploit reference information to guide realistic and semantically consistent reconstruction. Beyond reference inpainting, different LoRA-based methods [13, 31, 39] explore multi-concept composition.

Paint-by-Example (PbE) [51] replaces textual prompts with exemplar-driven diffusion conditioning through global image representations. TransFill [61] focuses on geometric alignment by generating multiple homography-based proposals to warp reference content into the target region. CorrFill [24] enforces correspondence-constrained diffusion guidance, which relies on accurate geometric matching under viewpoint changes. FaithFill [28] reports strong fidelity with efficient Low-Rank Adaptation (LoRA) fine-tuning. GeoComplete [21] injects explicit 3D structure and uses target-aware masking. RealFill [44] trains a diffusion inpainting model with both target and reference images to let the model learn patterns from a particular scene. LeftRefill [2] stitches the target and reference image together as the input to utilize the ability in a diffusion model. MimicBrush [6] injects reference-aware attention features through a dual UNet diffusion architecture. CompleteMe [45] similarly employs an additional UNet to extract detailed reference features, targeting human image completion.

However, these methods either underutilize the reference image or incur substantial computational overhead. Some approaches [24, 44, 51, 61] rely on global feature conditioning or high-quality geometric matching to guide generation. Others [6, 17, 45] trade performance by introducing additional modules, which

increase extra architectural complexity. In contrast, our method achieves effective reference-guided synthesis while keeping the architecture simple by reusing the same UNet backbone in two passes.

3 Method

We first formulate the problem in Sec. 3.1, then introduce the proposed framework in Sec. 3.2 along with the associated data construction in Sec. 3.3. Finally, we extend the framework to the multi-reference setting in Sec. 3.4.

3.1 Preliminaries

Reference-based inpainting aims to restore missing regions of a target image using additional appearance cues from reference images. Given an original image I (*i.e.*, also known as the ground-truth image) and a binary mask $M \in \{0,1\}^{H\times W}$, where $M = 1$ denotes missing pixels, the target image is

$$I_{tar} = I \odot (\mathbf{1} - M) + M, \tag{1}$$

where $\odot$ denotes element-wise multiplication and the missing region is filled with white ($M = 1$). Let $\{I_{ref_1}, \ldots, I_{ref_k}\}$ denote reference images, where typically $k < 5$. The goal is to synthesize a completed image as close as possible to the original image, preserving the visible context of the target while borrowing semantically compatible details from the references. Unless otherwise specified, we use the default single-reference setting ($k = 1$) for clarity.

3.2 Method Framework

An overview of the proposed framework is shown in Fig. 1. Our method builds upon a pre-trained Stable Diffusion (SD) v2 inpainting model [34] and introduces a *two-pass conditioning scheme* that reuses the same UNet twice.

Given a target image and a reference image, we first perform an auxiliary forward pass with an image set $\mathbf{I}_1$ containing the *reference* image. This pass, denoted as UNet-I, extracts reference-based features that serve as additional conditioning signals for generation. Next, a second pass with $\mathbf{I}_2$, containing the *target* image, is executed. The second pass, denoted as UNet-II, extracts target's background contextual features and performs the standard denoising process.

Both UNet-I and UNet-II share the same backbone and the same parameters. In the implementation, UNet-I retains only the down-sampling pathway for multi-level feature extraction, while UNet-II remains the full denoising network. Because the two passes are produced by identical weights, the features from UNet-I reside in the same feature space as those of UNet-II, and naturally match the spatial and channel dimensions of the corresponding layers. This property allows the reference features to be directly added into UNet-II to form

a unified conditioning signal, so that generation is guided by both *i*) appearance cues from the reference image and *ii*) reliable background context from the target image.

Existing architecture-augmented methods [6, 17, 45, 56], such as ControlNet and CompleteMe, attach an additional branch or UNet to extract conditioning features, and rely on extra modules to inject these features into the backbone (e.g., zero-convolutions or cross-attention). In contrast, our two passes share identical weights, ensuring that the extracted features naturally reside in a compatible representation space. This enables direct feature reuse through simple element-wise addition without any additional injection module, while preserving full compatibility with the original diffusion process.

3.3 Two-stage Data Construction

To support the two UNets passes framework, we design a two-stage data formulation that explicitly establishes correspondence at both image and region levels. The input data format is critical, as it forms the foundation for subsequent steps such as locating corresponding regions and extracting reference content.

Stage 1: Image-level Pairing. In this stage, we aim to build a unique input formulation for both passes to avoid ambiguity for the same UNet. We use $\mathbf{I}_1$ and $\mathbf{I}_2$ denote the input image set of UNet-I and UNet-II, respectively. The input $\mathbf{I}_1$ and $\mathbf{I}_2$ share the same format as follows:

$$\mathbf{I}_1 = \{I_{tar}, \mathbf{0}, I_{ref}\}, \mathbf{I}_2 = \{I_{tar},\ I_{tar}^{mask},\ I_{tar}^{cond}\}, \tag{2}$$

where $\mathbf{I}_2$ follows the standard input format of a Stable Diffusion inpainting model, as UNet-II corresponds to the original inpainting pass. Specifically, $\mathbf{I}_2$ consists of three components: the target image I_{tar}, the corresponding mask image I_{tar}^{mask}, and the conditioning target image I_{tar}^{cond}. Specifically, the effective mask is $I_{tar}^{mask} = (\mathbf{1} - M) \odot I_{mask}$, where I_{mask} is a randomly generated mask for training. $I_{tar}^{cond} = I \odot (\mathbf{1} - M) \odot (\mathbf{1} - I_{mask})$, which preserves only the known pixels. Similarly, $\mathbf{I}_1$ is constructed using the same input structure, consisting of three images: the target image I_{tar}, a zero mask image $\mathbf{0}$, and the reference image I_{ref}.

Since UNet-I is used to extract reference-conditioned features before the main denoising pass, we replace the second indicator mask image in $\mathbf{I}_1$ with a zero mask $\mathbf{0}$ (*i.e.*, a black image) to avoid introducing an explicit mask indicator in the first pass. This keeps the input interface of $\mathbf{I}_1$ consistent with the standard SD inpainting format, while ensuring that the first pass is primarily driven by the reference image to extract features. The role of the third image I_{ref} is further refined in Stage 2. Instead of feeding an explicit mask to UNet-I, we use the aligned and masked reference image to provide region-level guidance.

With this design, $\mathbf{I}_1$ and $\mathbf{I}_2$ share the same three-image input format, namely, a target image, a mask indicator, and a conditioning image. This unified interface follows the standard SD inpainting input formulation and allows the same pretrained UNet weights to be reused in both passes without ambiguity. Importantly, the two inputs emphasize complementary conditions: $\mathbf{I}_2$ preserves reliable

target background context, whereas $\mathbf{I}_1$ provides reference appearance for feature extraction. Thus, feeding $\{\mathbf{I}_1, \mathbf{I}_2\}$ as paired inputs establishes an explicit image-level correlation.

Stage 2: Region-level Correspondence. In Stage 2, we establish explicit region-level correspondences between the reference and the target images to provide structured conditioning signals for generation. In the standard SD inpainting formulation, the third input image serves as a conditioning image (e.g., I_{ref} in $\mathbf{I}_1$ and I_{tar}^{cond} in $\mathbf{I}_2$). Since I_{tar}^{cond} already preserves the reliable background content of the target image, we redesign the reference conditioning image I_{ref} in $\mathbf{I}_1$ to complement this information.

A key challenge is to incorporate both target background context and reference appearance cues within one forward process without introducing interference. To this end, we enforce spatial complementarity between the two conditioning images. Specifically, we encourage the reference image to provide candidate content only inside the masked region, while the target conditioning image supplies reliable context outside the masked region. This construction involves the following two steps:

i) Geometric alignment. Before constructing complementary masks, we first reduce geometric discrepancies between the reference and target images (e.g., viewpoint changes, scale variations, and perspective differences). An off-the-shelf feature matcher (e.g., OmniGlue [16]) is employed to estimate reliable sparse keypoint correspondences. Based on these correspondences, a coarse geometric transformation is estimated to warp the reference image, producing $\tilde{I}_{ref}$, such that informative regions approximately located within the masked area of the target image.

ii) Complementary masking. After obtaining the aligned reference image $\tilde{I}_{ref}$, we construct spatially complementary conditioning images as follows:

$$I_{\text{tar}}^{\text{cond}} = I'_{tar} \odot (\mathbf{1} - I_{mask}), \qquad I_{\text{ref}}^{\text{cond}} = \tilde{I}_{ref} \odot I_{mask}, \tag{3}$$

where $I'_{tar} = I \odot (\mathbf{1} - M)$ with missing region filled in black. Intuitively, $I_{\text{tar}}^{\text{cond}}$ preserves reliable background context from the target image, while $I_{\text{ref}}^{\text{cond}}$ supplies candidate content within the missing region from the reference image. Because the two conditioning images are constructed with complementary masks, their sum forms a coarse composite prior:

$$I_{\text{comp}}^{\text{cond}} = I_{\text{tar}}^{\text{cond}} + I_{\text{ref}}^{\text{cond}}. \tag{4}$$

Finally, we update $I_{ref} = I_{ref}^{cond}$ in Eq. (2), resulting in the revised inputs:

$$\mathbf{I}_1 = \{I_{tar}, \mathbf{0}, I_{ref}^{cond}\}, \quad \mathbf{I}_2 = \{I_{tar}, I_{tar}^{mask}, I_{tar}^{cond}\}. \tag{5}$$

Importantly, precise pixel-level alignment is not required. Since both conditioning images are encoded into latent representations and fused within the diffusion model, it is sufficient that semantically corresponding regions are approximately aligned within the masked area. This design ensures robustness to moderate geometric misalignment.

Once the two complementary inputs are constructed, they can be directly used in training. In each iteration, a random binary mask I_{mask} is sampled and used to construct the conditioning images I_{tar}^{cond} and I_{ref}^{cond}. The resulting inputs $\mathbf{I}_1$ and $\mathbf{I}_2$ are then encoded by the pretrained VAE $\mathcal{E}(\cdot)$, following the standard Stable Diffusion inpainting pipeline. Gaussian noise is subsequently added to the target latent $\mathcal{E}(I_{tar})$ to obtain the noisy latent z_t at step t, which is shared by both passes. In practice, UNet-I takes $\mathbf{I}_1$ to extract reference-dependent features, which are directly injected into the corresponding down-sampling layer of UNet-II by element-wise addition. Meanwhile, UNet-II takes $\mathbf{I}_2$ as the standard inpainting input and predicts the denoising target guided by the injected reference features. No additional scaling, normalization, or timestep-dependent scheduling is applied to the injected features.

Overall, the proposed two-stage data construction explicitly couples the target and reference images at two complementary levels: image-level pairing and region-level correspondence through complementary conditioning masks. This explicit spatial structuring provides correspondence cues for effective reference-guided diffusion synthesis, while remaining fully compatible with the standard inpainting formulation.

3.4 Inpainting with Multiple References

Our framework can be naturally extended to the multi-reference setting. When multiple reference images are available, we increase the number of forward passes of UNet-I according to the number of references k (i.e., UNet-I is executed k times for k references). Consequently, k reference-dependent latent features are extracted from UNet-I.

To remain compatible with the single-reference setting, the feature injected into UNet-II must preserve the same spatial size and channel dimensions as a single-reference feature. At the same time, we would like this injected feature to incorporate information from all available references. To this end, we treat I_{ref_1} as the primary reference and use a self-attention module to aggregate information from the remaining references into its representation.

As illustrated on the right-hand side of Fig. 1, the k latent features from different reference images are first concatenated along the channel dimension to form a joint feature, and then processed jointly by a self-attention module. This allows information from multiple references to interact within a shared representation. We then retain only the updated feature corresponding to the primary reference I_{ref_1} and inject it into UNet-II.

In this way, the injected feature has exactly the same shape as in the single-reference setting, while its representation is enriched by additional references (e.g., $I_{\mathrm{ref}_2}, I_{\mathrm{ref}_3}, I_{\mathrm{ref}_4}$) through the attention-based aggregation. This design allows additional references to contribute useful visual information to the injected feature, while keeping the injection mechanism unchanged.

4 Experiment

4.1 Dataset

Following LeftRefill [2], we adopt MegaDepth [20] dataset to construct image pairs. MegaDepth contains images of the same scenes captured under diverse viewpoints and illumination conditions, making it suitable for reference-guided image inpainting. Following LeftRefill [2], we select image pairs with 40% to 70% overlap as target and reference image pairs, and construct the evaluation split on the MegaDepth dataset by reserving 100 target images whose corresponding ground truth region are disjoint from the training set. Each target image is associated with four reference images, and all methods are evaluated using the same target-reference pairs. For mask generation, we first apply OmniGlue [16] to obtain sparse correspondences between the target and reference images. A random mask is then sampled within the overlapping region, increasing the chance that the reference image contains relevant content for completing the missing region in the target image. All images are resized to 512×512 resolution.

4.2 Baselines and Evaluation Metrics

We compare UNet-Twice with several representative image inpainting methods, including BrushNet [17], MimicBrush [6], Paint-by-Example (PbE) [51], CorrFill [24], RealFill [44], LeftRefill [2], and CompleteMe [45]. Among them, CorrFill is a plug-in correspondence module designed to be integrated with a base inpainting model rather than used independently. Following the official implementation, we evaluate CorrFill when plugged into Paint-by-Example, reported as "Paint-by-Example+CorrFill". BrushNet is a non-reference-based inpainting method, and CompleteMe is primarily designed for human image inpainting. We include them to provide a more comprehensive comparison. For each baseline, we follow the training and inference protocols described in the original papers or use the official pretrained checkpoints. Scene-specific methods (e.g., RealFill and UNet-Twice) follow the protocol of RealFill, where the model is trained separately for each scene on $\{I_{tar}, I_{ref}\}$ without access to the ground-truth region. Full-dataset training methods (e.g., LeftRefill and CompleteMe) are trained once on the full training split of approximately $499,000$ image pairs, together with $\{I_{tar}, I_{ref}\}$ of all test scenes, also without access to the ground-truth region. All methods are evaluated on the same target images, masks, and reference pairs.

The results are evaluated using Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) [48], Learned Perceptual Image Patch Similarity (LPIPS) [57], DreamSim [12], and DINO similarity [29]. PSNR and SSIM measure pixel-level reconstruction fidelity, LPIPS evaluates perceptual similarity, while DreamSim and DINO assess high-level semantic consistency between the result and ground-truth images. All metrics are computed after compositing the inpainted region back into the ground-truth image.

Table 1: Quantitative comparison under the single-reference (1-reference) inpainting setting. Following the official implementation, we evaluate CorrFill when plugged into Paint-by-Example, which is reported as "Paint-by-Example+CorrFill". The best results are highlighted in **bold**, and the second-best results are underlined.

Method	Low-level Quality			Semantic Similarity	
	PSNR ↑	SSIM ↑	LPIPS ↓	DreamSim ↓	DINO ↑
BrushNet	23.3474	0.9096	0.0610	0.0290	0.9568
Paint-by-Example (PbE)	23.3052	0.9103	0.0608	0.0311	0.9641
Paint-by-Example (PbE)+CorrFill	23.7351	0.9112	0.0594	0.0332	0.9667
MimicBrush	25.7770	0.9244	0.0464	0.0162	0.9798
CompleteMe	24.5790	0.9199	0.0506	0.0212	0.9765
RealFill	25.4339	0.9205	0.0500	0.0181	0.9816
LeftRefill	26.0812	0.9235	0.0475	0.0167	0.9807
Our (UNet-Twice)	**27.3856**	**0.9319**	**0.0412**	**0.0106**	**0.9874**

4.3 Implementation Details

Our framework is built upon the pre-trained Stable Diffusion v2 Inpainting model [34]. The default frozen CLIP text encoder [32] and VAE image encoder [34] are adopted following the original configuration. We freeze the entire down-sampling path and the mid-block, and only update the up-sampling layers. Accordingly, the features extracted by UNet-I and UNet-II (only the down-sampling stage) are always produced by frozen weights, while the updated weights are used in the up-sampling stage of UNet-II. Following the scene-specific protocol of RealFill [44], we optimize UNet-Twice for 2,000 iterations with a batch size of 8 for each scene. The learning rate is 2×10^{-5} with the Adam optimizer. n is set to 1, which is the number of injected features (*i.e.*, connected layers) from UNet-I to UNet-II. During inference, we follow prior works and employ the DDIM sampler [41] with 50 sampling steps. Detailed implementation settings, including training configurations, training and inference cost, are provided in the supplementary material.

4.4 Quantitative and Qualitative Comparison

Single-reference Inpainting We compare UNet-Twice with existing reference-based inpainting methods under the single-reference setting in Tab. 1. Overall, UNet-Twice achieves the best performance across all reported metrics, with notably improved low-level quality scores. These results suggest that the proposed framework better leverages reference information for faithful content reconstruction. Compared with approaches that rely on implicit correspondence discovery (e.g., attention-based matching or additional UNet branches), UNet-Twice explicitly structures reference guidance via approximate geometric alignment and complementary masking, which helps preserve reference-consistent details within the masked region.

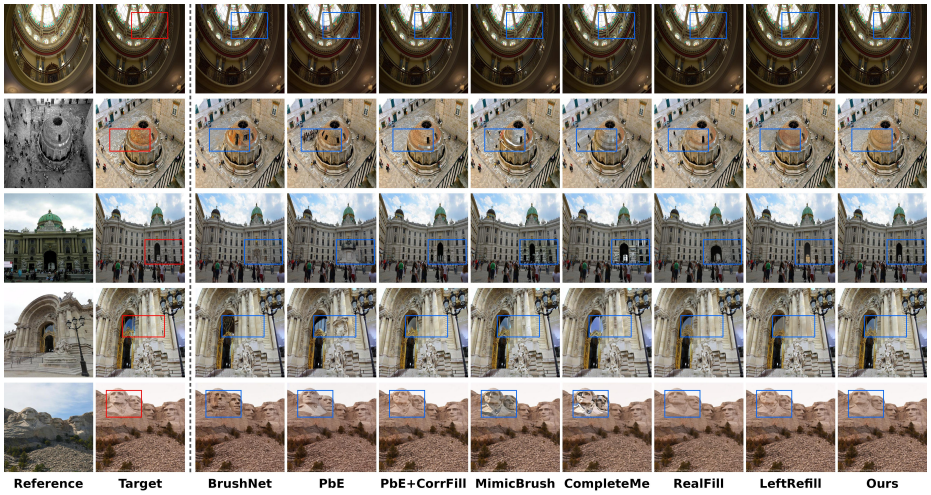

Fig. 2: Comparison of baselines on single-reference setting. Please refer to the red and blue boxes for a detailed comparison. PbE denotes Paint-by-Example.

Table 2: Quantitative comparison under multi-reference setting. The best results are highlighted in **bold**, and the second-best results are underlined.

Setting	Method	Low-level Quality			Semantic Similarity	
		PSNR ↑	SSIM ↑	LPIPS ↓	DreamSim ↓	DINO ↑
2-Refs	RealFill	26.2415	0.9229	0.0461	0.0123	0.9847
	LeftRefill	26.2392	0.9272	0.0423	0.0122	0.9854
	UNet-Twice	**28.4582**	**0.9393**	**0.0355**	**0.0077**	**0.9915**
3-Refs	RealFill	26.4494	0.9250	0.0438	0.0116	0.9874
	LeftRefill	26.3340	0.9274	0.0434	0.0142	0.9858
	UNet-Twice	**28.4586**	**0.9390**	**0.0355**	**0.0080**	**0.9910**
4-Refs	RealFill	26.6568	0.9263	0.0430	0.0113	0.9884
	LeftRefill	26.6212	0.9315	0.0455	0.0134	0.9868
	UNet-Twice	**28.5596**	**0.9341**	**0.0352**	**0.0081**	**0.9912**

Qualitative comparisons are shown in Fig. 2. While most baselines produce visually realistic results, they often fail to preserve fine details from the reference image or generate structures that are inconsistent with the reference appearance. In contrast, UNet-Twice more consistently preserves fine details and yields results that better align with both the target context and the reference appearance.

Multi-reference Inpainting. We compare UNet-Twice with RealFill and LeftRefill under the multi-reference setting in Tab. 2. Overall, UNet-Twice achieves the best performance across different numbers of reference images. Using multi-

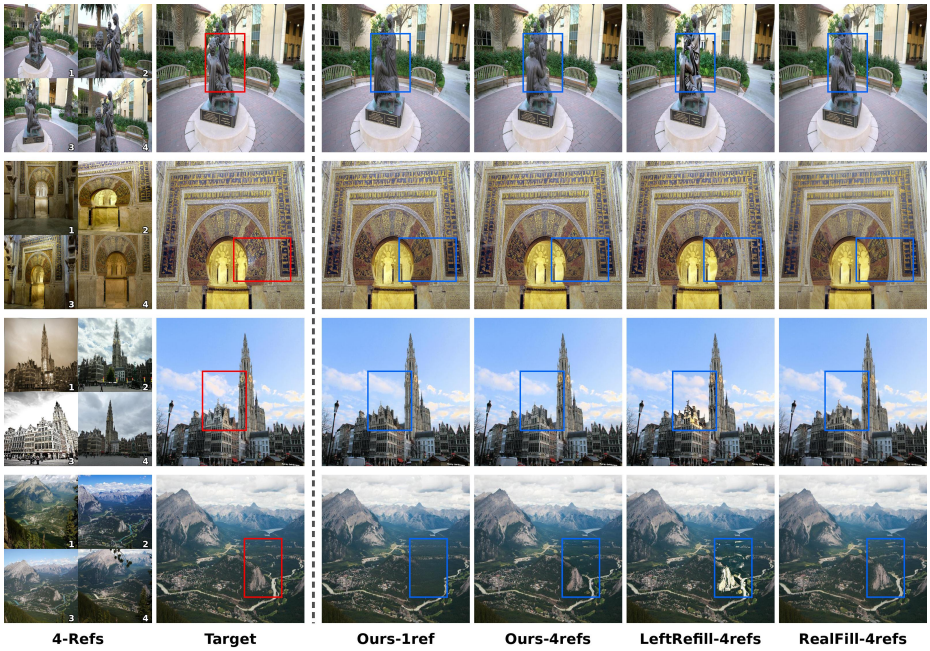

Fig. 3: Comparison under multi-reference setting. The figure shows: (i) results of UNet-Twice with 1-reference and 4-reference setting, and (ii) comparison between UNet-Twice, LeftRefill, and RealFill under the 4-reference setting.

ple references generally provides richer visual content than the single-reference setting, resulting in noticeable improvements across all metrics.

Qualitative comparisons are shown in Fig. 3. The figure includes two comparisons: (i) results of UNet-Twice under the 1-reference and 4-reference settings, and (ii) comparisons between UNet-Twice, RealFill, and LeftRefill under the 4-reference setting. For (i), UNet-Twice under the 4-reference setting shows clear improvements over the 1-reference setting, particularly when the primary reference does not sufficiently cover the missing region. For (ii), while all methods produce plausible results by leveraging multiple references, UNet-Twice better preserves fine details and yields more coherent structures in the restored regions.

4.5 Ablation Study

Number of Injected Features. Let n denote the number of feature tensors extracted from the down-sampling path of UNet-I and injected into UNet-II, *i.e.*, the number of connected layers between the two passes. In our UNet, the down-sampling stage consists of four down blocks with three layers each, together with a mid block, thus $n \in \{1, \ldots, 13\}$. The injection starts from the bottleneck and expands outward: $n = 1$ injects only the mid-block feature at the deepest position, and as n increases, additional features from shallower down-sampling

Table 3: Ablation study on different numbers of injected features n. n denotes the number of injected features from the down-sampling layers of UNet-I, which corresponds to the number of connected layers between UNet-I and UNet-II. The best results are highlighted in **bold**, and the second-best results are underlined.

Injected Features	Low-level Quality			Semantic Similarity	
	PSNR ↑	SSIM ↑	LPIPS ↓	DreamSim ↓	DINO ↑
n=1	**27.3858**	**0.9328**	**0.0412**	**0.0106**	**0.9874**
n=2	<u>27.1613</u>	0.9311	<u>0.0433</u>	0.0166	0.9809
n=3	27.1112	<u>0.9312</u>	<u>0.0433</u>	<u>0.0139</u>	0.9814
n=4	26.9773	0.9295	0.0440	0.0147	<u>0.9816</u>
n=7	26.9828	0.9293	0.0443	0.0147	0.9810
n=13	26.9106	0.9277	0.0451	0.0141	0.9811

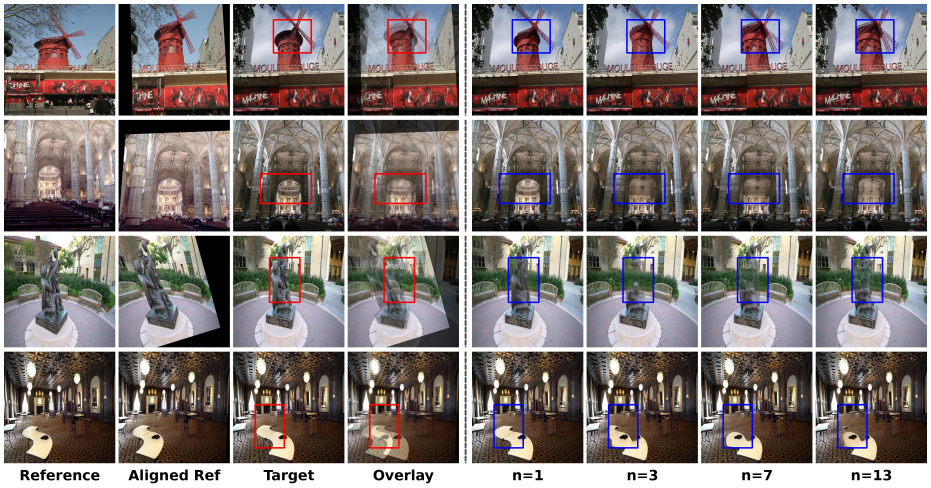

Fig. 4: Comparison under different numbers of injected features n. When the alignment is inaccurate, using a smaller n produces more realistic and semantically consistent results.

layers are progressively included. Consequently, a small n corresponds to deep, high-level semantic features, whereas a larger n additionally brings in shallower, low-level appearance-specific features.

Quantitative results are reported in Tab. 3. We observe that a smaller n consistently leads to better performance, and we therefore set $n = 1$ in all experiments. A likely reason is that when n is large (*e.g.*, $n = 13$), the injected low-level appearance dominates the denoising process, resulting in less realistic or less semantically consistent inpainting.

This phenomenon can be further visualized in Fig. 4. When $n = 13$, the content in the masked region is strongly influenced by the reference image. In some cases, the masked region looks as if the reference content were pasted in,

Table 4: Ablation study on different alignment methods. *-base* presents the default Homography alignment implemented in our framework. *-simple* is the version delete the second phase of fine alignment, and *-fine* is the version with a fine-designed second phase. *Affine* represents affine transformation. The best results are highlighted in **bold**, and the second-best results are underlined.

Alignment Method	Low-level Quality			Semantic Similarity	
	PSNR ↑	SSIM ↑	LPIPS ↓	DreamSim ↓	DINO ↑
Homography-base	27.3855	**0.9319**	0.0412	0.0106	0.9874
Homography-simple	**27.4491**	**0.9319**	0.0410	**0.0105**	**0.9878**
Homography-fine	27.4300	0.9318	**0.0409**	0.0108	0.9872
Affine	27.2807	0.9314	0.0410	0.0106	0.9870

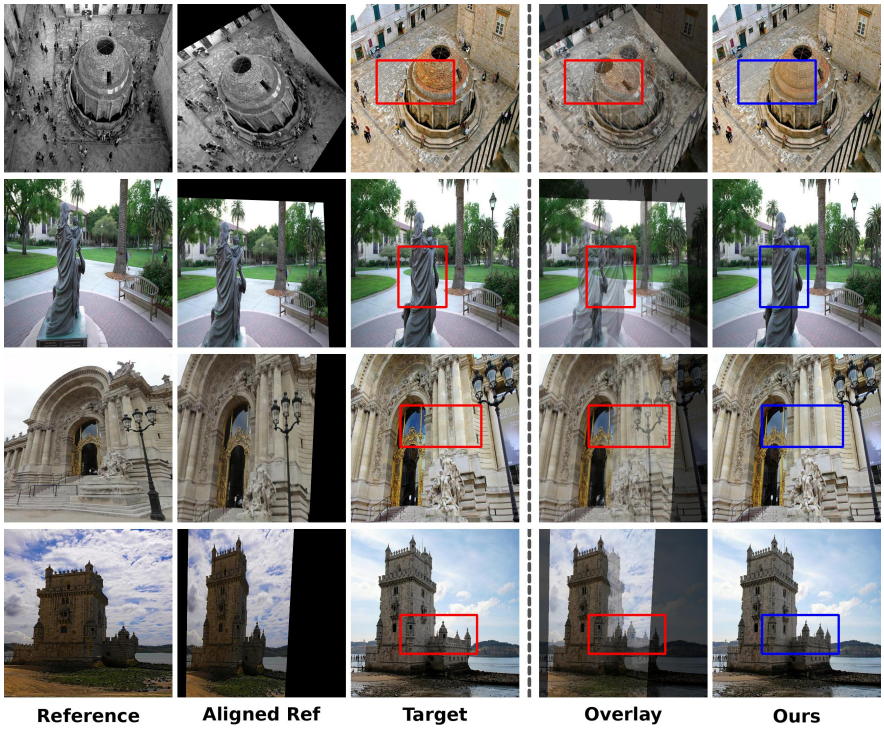

Fig. 5: Visualization results under imperfect geometric alignment. Even when the reference image cannot be precisely aligned with the target, UNet-Twice is still able to synthesize realistic and semantically coherent inpainting results.

rather than synthesized coherently. When the alignment between the two images is not very accurate, the discrepancy becomes visually obvious. This is consistent with the design of our framework: with a small n, only higher-level features are injected, making the method less sensitive to precise image alignment.

Alignment Strategies. Since the MegaDepth dataset contains a large number of architectural scenes, homography alignment provides a reasonable geometric approximation and is therefore adopted as the default. To evaluate whether our framework relies on highly refined alignment, we design three homography variants with increasing levels of refinement and compare them with the affine transformation. As shown in Tab. 4, varying the level of homography refinement leads to only minor performance differences. Moreover, although affine transformation is theoretically less suitable for architectural scenes due to its inability to model perspective projection, it results in only a marginal performance drop. These results indicate that our framework is robust to the choice of geometric alignment strategy.

Qualitative examples are shown in Fig. 5. It can be observed that when the aligned reference is not perfectly registered with the target, the generated results remain visually coherent and consistent with the reference content. This finding is consistent with the design of our framework: rather than relying on pixel-accurate correspondence, it is often sufficient for the semantically relevant regions of the reference to be approximately positioned within the masked area. This property helps relax the requirement on precise correspondence estimation.

5 Conclusion

In this paper, we propose UNet-Twice, a simple and elegant framework for reference-based image inpainting. UNet-Twice reuses the *same* UNet twice within a single forward process, avoiding the need for additional model branches. This design allows reference information to be naturally extracted and injected into a standard image inpainting process. We further introduce a structured paired input formulation that establishes explicit connections between the reference image and the target image at both the image and region levels. This design provides complementary conditioning features from both the reference and target image. Such complementary conditioning enables effective utilization of reference features and provides clear spatial guidance for the diffusion model during generation. Moreover, the proposed framework remains robust under coarse geometric alignment, reducing the dependence on precise alignment strategies. Extensive experiments demonstrate that UNet-Twice consistently outperforms existing methods in both single-reference and multi-reference settings across multiple evaluation metrics.

Acknowledgements

This work was supported by Western Sydney University through CDMS Departmental Funding (20860/04914) and Research Startup Funds (20860/86822), National Natural Science Foundation of China (No. 62476053), the Green Buildings Innovation Cluster 2.0 Challenge Call for Decarbonisation (GBIC R&D/2025/4).

References

1. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: Patchmatch: A randomized correspondence algorithm for structural image editing. ACM Trans. Graph. **28**(3), 24 (2009)
2. Cao, C., Cai, Y., Dong, Q., Wang, Y., Fu, Y.: Leftrefill: Filling right canvas based on left reference through generalized text-to-image diffusion model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7705–7715 (2024)
3. Chang, H., Zhang, H., Barber, J., Maschinot, A., Lezama, J., Jiang, L., Yang, M.H., Murphy, K., Freeman, W.T., Rubinstein, M., et al.: Muse: Text-to-image generation via masked generative transformers. arXiv preprint arXiv:2301.00704 (2023)
4. Chang, H., Zhang, H., Jiang, L., Liu, C., Freeman, W.T.: Maskgit: Masked generative image transformer. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11315–11325 (2022)
5. Chen, G., Zhang, G., Yang, Z., Liu, W.: Multi-scale patch-gan with edge detection for image inpainting: Multi-scale patch-gan with edge detection for image inpainting. Applied intelligence **53**(4), 3917–3932 (2023)
6. Chen, X., Feng, Y., Chen, M., Wang, Y., Zhang, S., Liu, Y., Shen, Y., Zhao, H.: Zero-shot image editing with reference imitation. Advances in Neural Information Processing Systems **37**, 84010–84032 (2024)
7. Chen, X., Feng, Y., Chen, M., Wang, Y., Zhang, S., Yu, L., Shen, Y., Zhao, H.: Zero-shot image editing with reference imitation. arXiv preprint arXiv:2406.07547 (2024)
8. Chen, X., Huang, L., Liu, Y., Shen, Y., Zhao, D., Zhao, H.: Anydoor: Zero-shot object-level image customization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6593–6602 (2024)
9. Criminisi, A., Pérez, P., Toyama, K.: Region filling and object removal by exemplar-based image inpainting. IEEE Transactions on image processing **13**(9), 1200–1212 (2004)
10. Demir, U., Unal, G.: Patch-based image inpainting with generative adversarial networks. arXiv preprint arXiv:1803.07422 (2018)
11. Deng, Y., Hui, S., Zhou, S., Meng, D., Wang, J.: T-former: An efficient transformer for image inpainting. In: Proceedings of the 30th ACM international conference on multimedia. pp. 6559–6568 (2022)
12. Fu, S., Tamir, N., Sundaram, S., Chai, L., Zhang, R., Dekel, T., Isola, P.: Dreamsim: Learning new dimensions of human visual similarity using synthetic data. arXiv preprint arXiv:2306.09344 (2023)
13. Gu, Y., Wang, X., Wu, J.Z., Shi, Y., Chen, Y., Fan, Z., Xiao, W., Zhao, R., Chang, S., Wu, W., et al.: Mix-of-show: Decentralized low-rank adaptation for multi-concept customization of diffusion models. Advances in Neural Information Processing Systems **36**, 15890–15902 (2023)
14. Hedjazi, M.A., Genc, Y.: Efficient texture-aware multi-gan for image inpainting. Knowledge-Based Systems **217**, 106789 (2021)
15. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
16. Jiang, H., Karpur, A., Cao, B., Huang, Q., Araujo, A.: Omniglue: Generalizable feature matching with foundation model guidance. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19865–19875 (2024)

17. Ju, X., Liu, X., Wang, X., Bian, Y., Shan, Y., Xu, Q.: Brushnet: A plug-and-play image inpainting model with decomposed dual-branch diffusion. In: European Conference on Computer Vision. pp. 150–168. Springer (2024)
18. Ko, K., Kim, C.S.: Continuously masked transformer for image inpainting. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 13169–13178 (2023)
19. Li, W., Lin, Z., Zhou, K., Qi, L., Wang, Y., Jia, J.: Mat: Mask-aware transformer for large hole image inpainting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10758–10768 (2022)
20. Li, Z., Snavely, N.: Megadepth: Learning single-view depth prediction from internet photos. In: Computer Vision and Pattern Recognition (CVPR) (2018)
21. Lin, B., Chen, T., Tan, R.T.: Geocomplete: Geometry-aware diffusion for reference-driven image completion. arXiv preprint arXiv:2510.03110 (2025)
22. Liu, G., Reda, F.A., Shih, K.J., Wang, T.C., Tao, A., Catanzaro, B.: Image inpainting for irregular holes using partial convolutions. In: Proceedings of the European conference on computer vision (ECCV). pp. 85–100 (2018)
23. Liu, H., Wan, Z., Huang, W., Song, Y., Han, X., Liao, J.: Pd-gan: Probabilistic diverse gan for image inpainting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9371–9381 (2021)
24. Liu, K.H., Yang, C.K., Chen, M.H., Liu, Y.L., Lin, Y.Y.: Corrfill: Enhancing faithfulness in reference-based inpainting with correspondence guidance in diffusion models. In: 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 1618–1627. IEEE (2025)
25. Liu, T., Liao, L., Chen, D., Xiao, J., Wang, Z., Lin, C.W., Satoh, S.: Transref: Multi-scale reference embedding transformer for reference-guided image inpainting. Neurocomputing **632**, 129749 (2025)
26. Liu, X., Park, D.H., Azadi, S., Zhang, G., Chopikyan, A., Hu, Y., Shi, H., Rohrbach, A., Darrell, T.: More control for free! image synthesis with semantic diffusion guidance. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 289–299 (2023)
27. Lugmayr, A., Danelljan, M., Romero, A., Yu, F., Timofte, R., Van Gool, L.: Repaint: Inpainting using denoising diffusion probabilistic models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11461–11471 (2022)
28. Mallick, R., Abdalla, A., Bargal, S.A.: Faithfill: Faithful inpainting for object completion using a single reference image. arXiv preprint arXiv:2406.07865 (2024)
29. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
30. Park, D.H., Luo, G., Toste, C., Azadi, S., Liu, X., Karalashvili, M., Rohrbach, A., Darrell, T.: Shape-guided diffusion with inside-outside attention. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 4198–4207 (2024)
31. Po, R., Yang, G., Aberman, K., Wetzstein, G.: Orthogonal adaptation for modular customization of diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7964–7973 (2024)
32. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning (ICML) (2021)

33. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:2204.06125 **1**(2), 3 (2022)
34. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10684–10695 (June 2022)
35. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)
36. Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., Aberman, K.: 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. pp. 22500–22510 (2023)
37. Saharia, C., Chan, W., Chang, H., Lee, C., Ho, J., Salimans, T., Fleet, D., Norouzi, M.: Palette: Image-to-image diffusion models. In: ACM SIGGRAPH 2022 conference proceedings. pp. 1–10 (2022)
38. Sargsyan, A., Navasardyan, S., Xu, X., Shi, H.: Mi-gan: A simple baseline for image inpainting on mobile devices. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7335–7345 (2023)
39. Shah, V., Ruiz, N., Cole, F., Lu, E., Lazebnik, S., Li, Y., Jampani, V.: Ziplora: Any subject in any style by effectively merging loras. In: European Conference on Computer Vision. pp. 422–438. Springer (2024)
40. Shamsolmoali, P., Zareapoor, M., Granger, E.: Transinpaint: Transformer-based image inpainting with context adaptation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 849–858 (2023)
41. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020)
42. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. arXiv preprint arXiv:2011.13456 (2020)
43. Sun, J., Shen, Z., Wang, Y., Bao, H., Zhou, X.: Loftr: Detector-free local feature matching with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8922–8931 (2021)
44. Tang, L., Ruiz, N., Chu, Q., Li, Y., Holynski, A., Jacobs, D.E., Hariharan, B., Pritch, Y., Wadhwa, N., Aberman, K., et al.: Realfill: Reference-driven generation for authentic image completion. ACM Transactions on Graphics (ToG) **43**(4), 1–12 (2024)
45. Tsai, Y.J., Price, B., Liu, Q., Figueroa, L., Pakhomov, D., Ding, Z., Cohen, S., Yang, M.H.: Completeme: Reference-based human image completion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 18252–18261 (2025)
46. Wang, W., Niu, L., Zhang, J., Yang, X., Zhang, L.: Dual-path image inpainting with auxiliary gan inversion. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11421–11430 (2022)
47. Wang, Y., Cao, C., Yu, J., Fan, K., Xue, X., Fu, Y.: Towards enhanced image inpainting: Mitigating unwanted object insertion and preserving color consistency. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 23237–23248 (2025)

48. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004)
49. Xie, L., Pakhomov, D., Wang, Z., Wu, Z., Chen, Z., Zhou, Y., Zheng, H., Zhang, Z., Lin, Z., Zhou, J., et al.: Turbofill: adapting few-step text-to-image model for fast image inpainting. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 7613–7622 (2025)
50. Xie, Z., Lai, X., Zhao, W., Jiang, S., Liu, X., Hou, W.: Modification takes courage: Seamless image stitching via reference-driven inpainting. arXiv preprint arXiv:2411.10309 (2024)
51. Yang, B., Gu, S., Zhang, B., Zhang, T., Chen, X., Sun, X., Chen, D., Wen, F.: Paint by example: Exemplar-based image editing with diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18381–18391 (2023)
52. Yi, Z., Tang, Q., Azizi, S., Jang, D., Xu, Z.: Contextual residual aggregation for ultra high-resolution image inpainting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7508–7517 (2020)
53. Zeng, Y., Fu, J., Chao, H., Guo, B.: Aggregated contextual transformations for high-resolution image inpainting. IEEE transactions on visualization and computer graphics **29**(7), 3266–3280 (2022)
54. Zeng, Y., Lin, Z., Yang, J., Zhang, J., Shechtman, E., Lu, H.: High-resolution image inpainting with iterative confidence feedback and guided upsampling. In: European conference on computer vision. pp. 1–17. Springer (2020)
55. Zhang, D., Liang, Z., Yang, G., Li, Q., Li, L., Sun, X.: A robust forgery detection algorithm for object removal by exemplar-based image inpainting. Multimedia Tools and Applications **77**(10), 11823–11842 (2018)
56. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3836–3847 (2023)
57. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018)
58. Zhao, H., Zeng, Y., Lu, H., Wang, L.: Large occluded human image completion via image-prior cooperating. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 7469–7477 (2024)
59. Zhao, W., Rao, Y., Liu, Z., Liu, B., Zhou, J., Lu, J.: Unleashing text-to-image diffusion models for visual perception. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5729–5739 (2023)
60. Zhao, Y., Barnes, C., Zhou, Y., Shechtman, E., Amirghodsi, S., Fowlkes, C.: Geofill: Reference-based image inpainting with better geometric understanding. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1776–1786 (2023)
61. Zhou, Y., Barnes, C., Shechtman, E., Amirghodsi, S.: Transfill: Reference-guided image inpainting by merging multiple color and spatial transformations. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2266–2276 (2021)