PixelMan: Consistent Object Editing with Diffusion Models via Pixel Manipulation and Generation

Liyao Jiang[1,2], **Negar Hassanpour**[2], **Mohammad Salameh**[2], **Mohammadreza Samadi**[2], **Jiao He**[3], **Fengyu Sun**[3], **Di Niu**[1]
[1]Department of Electrical and Computer Engineering, University of Alberta
[2]Huawei Technologies Canada
[3]Huawei Kirin Solution, China
{liyao1, dniu}@ualberta.ca
{negar.hassanpour2, mohammad.salameh, mohammadreza.samadi1, hejiao4}@huawei.com
sunfengyu@hisilicon.com

Abstract

Recent research explores the potential of Diffusion Models (DMs) for consistent object editing, which aims to modify object position, size, and composition, etc., while preserving the consistency of objects and background without changing their texture and attributes. Current inference-time methods often rely on DDIM inversion, which inherently compromises efficiency and the achievable consistency of edited images. Recent methods also utilize energy guidance which iteratively updates the predicted noise and can drive the latents away from the original image, resulting in distortions. In this paper, we propose PixelMan, an inversion-free and training-free method for achieving consistent object editing via Pixel Manipulation and generation, where we directly create a duplicate copy of the source object at target location in the pixel space, and introduce an efficient sampling approach to iteratively harmonize the manipulated object into the target location and inpaint its original location, while ensuring image consistency by anchoring the edited image to be generated to the pixel-manipulated image as well as by introducing various consistency-preserving optimization techniques during inference. Experimental evaluations based on benchmark datasets as well as extensive visual comparisons show that in as few as 16 inference steps, PixelMan outperforms a range of state-of-the-art training-based and training-free methods (usually requiring 50 steps) on multiple consistent object editing tasks.

Project — liyaojiang1998.github.io/projects/PixelMan/
Extended version — https://arxiv.org/abs/2412.14283

Introduction

Diffusion Models (DMs) excel at generating stunning visuals from text prompts (Rombach et al. 2022; Saharia et al. 2022b; Chang et al. 2023), yet with potentials extending beyond text-to-image generation. A highly popular application is image editing, as evidenced by widespread tools such as Google Photos MagicEditor (Google 2023) and AI Editor in Adobe Photoshop (Adobe 2023). Many research efforts (Hertz et al. 2022; Tumanyan et al. 2023; Alaluf et al. 2023; Parmar et al. 2023) achieve promising results on text-prompt-guided rigid image editing involving tasks such as changing the color, texture, attributes, and style of the image. However, *consistent object editing* (Kawar et al. 2023; Cao et al. 2023; Duan et al. 2024) is a distinct type of image editing that aims to preserve the consistency of objects and background in the image without changing their texture and attributes, while modifying only certain non-rigid attributes of the objects (e.g., changing the position, size, and composition of objects). Typical consistent object editing tasks include object repositioning (Epstein et al. 2023; Mou et al. 2024b,a; Wang et al. 2024; Winter et al. 2024), object resizing (Epstein et al. 2023; Mou et al. 2024b,a), and object pasting (Chen et al. 2024; Mou et al. 2024b,a). Consistent object editing tasks are complex and usually involve multiple sub-tasks such as: (i) generating a faithful reproduction of the source object at the target location, (ii) maintaining the background scene details, (iii) harmonizing the edited object into its surrounding target context, and (iv) inpainting the original vacated location with a cohesive background.

To solve this problem, training-based methods have been proposed (Rombach et al. 2022; Chen et al. 2024; Wang et al. 2024; Winter et al. 2024), which however require a costly training process and usually also require collecting task-specific datasets. Alternatively, recent training-free methods (Epstein et al. 2023; Mou et al. 2024b,a) rely on DDIM inversion (Dhariwal and Nichol 2021) to estimate the initial noise corresponding to the source image. However, this process is inefficient as it often requires many (usually at least 50) inference steps. Reducing the number of steps to, e.g., 16, significantly compromises editing quality (see Fig. 2). Moreover, DDIM inversion struggles to produce a precise and consistent final reconstruction of the source image, often yielding a coarse approximation due to accumulation of errors at each timestep (Duan et al. 2024). As a result, training-free methods that rely on DDIM inversion are inherently limited in their ability to perform consistent edits.

To facilitate object generation at target location and reproduction of background, DragonDiffusion (Mou et al. 2024b) and DiffEditor (Mou et al. 2024a) utilize Energy Guidance (EG) to minimize the feature similarity between the source and target objects (backgrounds). While EG iteratively refines the predicted noise, this process can inadvertently drive the latent representation away from that of the

original image during inference, causing distortions in object appearance and background. Additionally, seamlessly inpainting the vacated region (if any) with a coherent background remains a challenge, as existing methods often struggle to fully remove the original object or introduce unintended elements (see Fig. 2).

In this paper, we propose PixelMan, an inversion-free and training-free method to achieve consistent object editing with existing pretrained text-to-image diffusion models via Pixel Manipulation and generation in as few as 16 steps that outperform all competitive training-based and training-free methods (usually requiring 50 steps) on a range of consistent object editing tasks. Rather than performing DDIM inversion and edited denoising, we directly create a duplicate copy of the source object at target location in the pixel space, and introduce an efficient sampling approach to iteratively harmonize the manipulated object into the target location and inpaint its original location, while ensuring image consistency by anchoring output image to be generated to the pixel-manipulated image as well as by introducing various consistency-preserving optimization techniques during inference. Our contributions are summarized as follows:

- We propose to perform pixel manipulation for achieving consistent object editing, by creating a pixel-manipulated image where we copy the source object to the target location in the pixel space. At each step, we always anchor the target latents to the pixel-manipulated latents, which reproduces the object and background with high image consistency, while only focusing on generating the missing "delta" between the pixel-manipulated image and the target image to be generated.
- We design an efficient three-branched inversion-free sampling approach, which finds the delta editing direction to be added on top of the anchor, i.e., the latents of the pixel-manipulated image, by computing the difference between the predicted latents of the target image and pixel-manipulated image in each step. This process also facilitates faster editing by reducing the required number of inference steps and number of Network Function Evaluations (NFEs).
- To inpaint the manipulated object's source location, we identify a root cause of many incomplete or incoherent inpainting cases in practice, which is attributed to information leakage from similar objects through the Self-Attention (SA) mechanism. To address this issue, we propose a leak-proof self-attention technique to prevent attention to source, target, and similar objects in the image to mitigate leakage and enable cohesive inpainting.
- Our method harmonizes the edited object with the target context, by leveraging editing guidance with latents optimization, and by using a source branch to preserve uncontaminated source K, V features as the context for generating appropriate harmonization effects (e.g. lighting, shadow, and edge blending) at the target location.

We provide extensive quantitative and/or qualitative visual comparisons to a range of state-of-the-art training-free and training-based approaches designed for object repositioning, object resizing and object pasting (some of which can be found in Appendix). Quantitative results on the COCOEE and ReS datasets as well as extensive visual comparisons suggest that PixelMan achieves superior performance in consistency metrics for object, background, and semantic consistency between the source and edited image, while achieving higher or comparable performance in IQA metrics. As a training-free method, PixelMan only requires 16 inference steps with lower average latency and a lower number of NFEs than current popular methods.

Related Works

Image editing with DMs. While standard text-to-image DMs are not directly designed for image editing, recent research is actively exploring their potential for this task. *Training-based* approaches (Saharia et al. 2022a; Brooks, Holynski, and Efros 2023; Zhang, Rao, and Agrawala 2023) optimize the UNet for certain editing scenarios. Wang et al. (2024) fine-tuned an inpainting model specifically for object repositioning task (by introducing and utilizing an ad-hoc dataset, namely ReS). However, these approaches may require high computational resources only to learn a specific task. As such, there is a high motivation to explore methods for augmenting pretrained UNets with different editing capabilities without additional training. In *training-free* methods (Hertz et al. 2022; Alaluf et al. 2023; Hertz et al. 2023; Tumanyan et al. 2023), users can perform editing either by a descriptive text prompt (Hertz et al. 2022; Brooks, Holynski, and Efros 2023; Tumanyan et al. 2023; Epstein et al. 2023), or by specifying editing points within an image, called *point-based* editing (Endo 2022; Pan et al. 2023; Shi et al. 2023; Mou et al. 2024b,a). The main advantage of point-based editing is the granular control over the edit region. In this work, we propose a point-based training-free approach for consistent object editing using DM, which preserves the consistency between the source and edited image.

Training-free consistent object editing. Epstein et al. (2023) introduced Energy Guidance (EG) (see Appendix for details) and proposed SelfGuidance, a prompt-based editing method that guides the sampling process based on specific energy functions defined on attentions and activations. Mou et al. (2024b) proposed DragonDiffusion, a point-based editing approach that leverages EG to update the sampled noise. Building on this, DiffEditor (Mou et al. 2024a) improved the content consistency by introducing regional SDE sampling and score-based gradient guidance (Song et al. 2020). Despite their success, EG-based methods require computationally expensive tricks to propagate the guidance from ϵ to z_t. Different from EG-based methods that update the estimated noise ϵ, our method directly updates the latents z_t for consistent object editing, which reduces the latency as well as NFEs, while maintaining consistency and image quality.

Inverting real images. Preserving the consistency between the original and edited image is crucial for consistent image editing. Training-free methods often utilize the inversion techniques to convert the source image into a convertible initial noise (z_T). DDIM inversion (Dhariwal and Nichol 2021) is a common but computationally expensive technique as it usually requires 50 inference steps.

ReNoise (Garibi et al. 2024) is a recent inversion technique that can utilize few-steps models (Luo et al. 2023; AI 2023), but its repeated UNet calls in its refinement phase still leads to high computation costs. An alternative approach is Denoising Diffusion Consistent Model (DDCM) (Xu et al. 2024), which facilitates inversion-free prompt-guided rigid image editing for changing the texture and attribute of objects. In contrast to DDCM, our method does not use any prompt, and instead we propose an inversion-free approach for efficient consistent object editing which focuses on preserving the consistency of objects and background in the image without changing their texture and attributes while modifying only certain non-rigid attributes of the objects (e.g., changing the position, size, and composition of objects).

Attention control for editing. Recent studies on training-free editing techniques (Cao et al. 2023; Hertz et al. 2023; Tumanyan et al. 2023; Hertz et al. 2022; Parmar et al. 2023) explore either integrating or manipulating Cross-Attentions (CAs) and Self-Attentions (SAs) to exert precise control over the editing process. Manipulating CAs has been demonstrated to offer control over object composition. Hertz et al. (2022) proposed an injection approach for swapping objects and changing the global style. Alternatively, since SAs incorporate information about pixel interactions in the spatial domain, manipulating them affects overall style, texture, and object interaction (Alaluf et al. 2023; Hertz et al. 2023; Jeong et al. 2024; Cao et al. 2023). Building on this, Patashnik et al. (2023) presented a SA injection method to selectively preserve a set of objects while altering other regions. Following these insights, we propose a leakproof self-attention technique to ensure a complete and cohesive inpainting of the vacated area with the background, by preventing a root cause of failed inpainting which is information leakage from the source or similar objects.

Method

To enhance computational efficiency and preserve image consistency during object editing, we introduce PixelMan, an efficient *inversion-free* and *training-free* method that performs *consistent object editing* with DMs via Pixel Manipulation and generation in *few inference steps*. The following subsections describe our proposed three-branched inversion-free sampling approach, leakproof self-attention technique, and editing guidance with latents optimization method.

Three-Branched Inversion-Free Sampling

Our goal is to achieve the following three objectives with high efficiency: (i) consistent reproduction of the object and background; (ii) object-background harmonization; and (iii) cohesive inpainting of the vacated location. As the backbone of PixelMan, we propose a three-branched sampling paradigm that achieves these three objectives using a single pretrained DM, while also bypassing inversion to facilitate faster editing by reducing inference steps and NFEs.

Specifically, we utilize three separate branches: *source branch*, *pixel-manipulated branch*, and *target branch*. Each branch maintains its own nosiy latents that is initialized, denoised (using the same UNet), and updated in different manners throughout the T sampling time-steps $t \in [1, T]$. We denote the noisy latents of the source branch, pixel-manipulated branch, and target branch at time-step t respectively with z_t^{src}, z_t^{man}, z_t^{tgt}.

We create a *pixel-manipulated image* I_{man} by copying the source object to the target location in pixel-space. For the object resizing task, we interpolate the object pixels based on the resizing scale before making a copy at the target location. For object pasting, the source object comes from a separate reference image I_{ref}, and is copied into the source image I_{src} at the target location to create I_{man}. Then, using the VAE encoder, we encode the pixel-manipulated image I_{man} and the source image I_{src} respectively into the pixel-manipulated latents z_0^{man} and source latents z_0^{src}.

Pixel-manipulated latents as anchor. At each time-step t of our sampling process, our goal is to directly obtain a latent space estimation of the edited output image I_{out} which we denote as output latents z_0^{out}. First, we ask the question of *what would be a reasonable estimate of the output latents* z_0^{out}. Intuitively, our estimation of output latents z_0^{out} should be identical to the source latents z_0^{src} so we can exactly reproduce the source image.

However, we want to reproduce the source object at the new target location, so we set our estimation of the output latents z_0^{out} to be identical to the pixel-manipulated latents z_0^{man}, which already have the source object reproduced at the target location through pixel manipulation. By using this naive estimation $z_0^{\text{out}} = z_0^{\text{man}}$, we can already effortlessly preserve the original background and consistently reproduce the object at the target location. Therefore, we refer to this pixel-manipulated latents z_0^{man} as the anchor.

In addition to image consistency, we also want to achieve cohesive inpaitning of the vacated location, and harmonize the object and background with realistic effects. So, there should be a delta editing direction Δz added on top of the anchor z_0^{man} to achieve the inpainting and harmonization edits. More concretely, at each time-step t, we set the output latent z_0^{out} as:

$$z_0^{\text{out}} = z_0^{\text{man}} + \Delta z. \tag{1}$$

With our simple estimation of output latents z_0^{out} using the sum of the anchor z_0^{man} and the delta edit direction Δz, we can preserve the object and background consistency without any inversion, which improves both the efficiency and image consistency by avoiding the computation bottleneck and accumulated reconstruction error of the DDIM inversion (Dhariwal and Nichol 2021) process. Next, we introduce our method for obtaining the delta edit direction Δz.

Obtaining delta edit direction. We aim to obtain the delta editing direction Δz that can achieve cohesive inpainting of the vacated location, and harmonize the object and background with realistic effects (e.g., lighting, shadow, edge blending). To achieve this, we propose to apply several editing guidance techniques (introduced in the later sections) for generating the inpainting and harmonization edits in the target branch, including leak-proof self-attention, editing guidance with latent optimization, and injection of source K, V features into the target branch. Meanwhile, we keep the

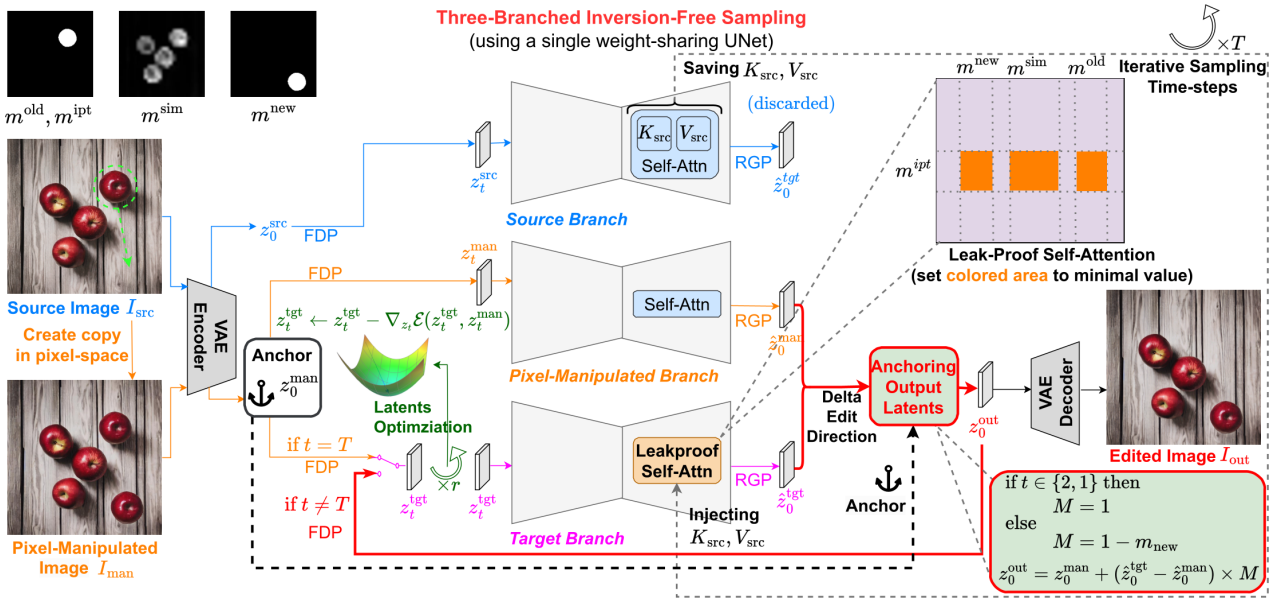


Figure 1: **Overview of PixelMan.** An efficient inversion-free sampling approach for consistent image editing, which copies the object to target location in pixel-space, and ensure image consistency by anchoring to the latents of pixel-manipulated image. We design a leak-proof self-attention mechanism to achieve complete and cohesive inpainting by mitigating information leakage.

pixel-manipulated branch consistent with the anchor z_0^{man}, and obtain Δz by finding the difference in the output of the two branches.

Specifically, we calculate the difference between the predicted target latents $\hat{z}_0^{\text{tgt}}$ from the target branch and predicted pixel-manipulated latents $\hat{z}_0^{\text{man}}$ from the pixel-manipulated branch:

$$\Delta z = \hat{z}_0^{\text{tgt}} - \hat{z}_0^{\text{man}}. \tag{2}$$

To obtain $\hat{z}_0^{\text{man}}$ from the **pixel-manipulated branch**, we always analytically compute the noisy latents z_t^{man} at each sampling time-step t from the pixel-manipulated latents z_0^{man} (i.e., the anchor that ensures consistency to I_{man}) which has already reproduced object at the target location and the original background.

Specifically, at each time-step t, we first follow the FDP equation to obtain z_t^{man} by adding random Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ to z_0^{src}:

$$z_t^{\text{man}} = \sqrt{\bar{\alpha}_t} \times z_0^{\text{man}} + \sqrt{1 - \bar{\alpha}_t} \times \epsilon. \tag{3}$$

Then, we pass the noisy source latents z_t^{man} to the denoising UNet (parameterized by θ) to get the predicted noise $\hat{\epsilon}_t^{\text{man}}$ at time-step t:

$$\hat{\epsilon}_t^{\text{man}} = \text{UNet}(z_t^{\text{man}}, t). \tag{4}$$

Finally, we obtain the predicted pixel-manipulated latents $\hat{z}_0^{\text{man}}$ based on the noisy pixel-manipulated latents z_t^{man} and the UNet predicted noise $\hat{\epsilon}_t^{\text{man}}$ at time-step t, using the Reverse Generative Process (RGP) (more details in Eq. (10) in the Appendix):

$$\hat{z}_0^{\text{man}} = \text{RGP}(z_t^{\text{man}}, \hat{\epsilon}_t^{\text{man}}, t). \tag{5}$$

Next, we obtain $\hat{z}_0^{\text{tgt}}$ from the **target branch**. Before the initial timestep $t = T$, we initialize the target latents z_0^{tgt} to be the same as z_0^{man} which corresponds to the anchor I_{man}. In contrast, at each sampling time-step t, we instead utilize the FDP similar to Eq. (3) to analytically compute the noisy target latents z_t^{tgt} from the estimated output latents z_0^{out} of previous time-step.

Then, z_t^{tgt} is updated with latents optimization (detailed in "Editing Guidance with Latents Optimization"). Next, we pass z_t^{tgt} along with the saved source branch K_{src}, V_{src} (detailed in "Feature-preserving source branch") to the UNet to obtain the predicted noise $\hat{\epsilon}_t^{\text{src}}$, where $\hat{\epsilon}_t^{\text{src}} = \text{UNet}(z_t^{\text{src}}, t; \{K_{\text{src}}, V_{\text{src}}\})$. Next, we obtain the predicted target latents $\hat{z}_0^{\text{tgt}}$ using the RGP similar to Eq. (5).

After calculating both $\hat{z}_0^{\text{tgt}}$ and $\hat{z}_0^{\text{man}}$, we finally obtain the delta editing direction Δz. To estimate the output image, we combine the anchor z_0^{man} and the delta editing direction in Eq. (2), while applying a masked-blending approach with mask $(1 - m_{\text{new}})$ (i.e., the object target location):

$$z_0^{\text{out}} = z_0^{\text{man}} + (\hat{z}_0^{\text{tgt}} - \hat{z}_0^{\text{man}}) \times (1 - m_{\text{new}}). \tag{6}$$

The masked-blending is applied throughout the sampling time-steps to remove the delta editing direction Δz in the target location, and only use the anchor z_0^{man} to achieve object consistency. While allowing Δz to change the background for inpainting and harmonization. For the last few time-steps no masking is applied, which encourages seamless object-background blending and allows the DM to refine the details of the output image.

Feature-preserving source branch. At each time-step t, we always analytically compute the noisy source latents z_t^{src} from z_0^{src} (making z_t^{src} consistent with z_0^{src}). Specifically, at each time-step t, we first follow the FDP equation similar to Eq. (3) to obtain z_t^{src} by adding random Gaussian noise

$\epsilon \sim \mathcal{N}(0, I)$ to z_0^{src}. Then, we pass the noisy source latents z_t^{src} to the denoising UNet to get the predicted noise $\hat{\epsilon}_t^{\text{src}}$ at time-step t: $\hat{\epsilon}_t^{\text{src}}, \{K_{\text{src}}, V_{\text{src}}\} = \text{UNet}(z_t^{\text{src}}, t)$. Note that the $\hat{\epsilon}_t^{\text{src}}$ is discarded here, and we save the self-attention K_{src} and V_{src} matrices from the source branch and inject[1] them back during the UNet call on z_t^{tgt} in the target branch, which is inspired by the mutual self-attention technique proposed in (Cao et al. 2023). The saved K_{src} and V_{src} preserve the original visual details from I_{src}, and the injection into target branch serves as context for generating appropriate harmonization effects (e.g., lighting, shadow, and edge blending), and also for inpainting the vacated area.

Leak-Proof Self-Attention

Our objective is to achieve complete and cohesive inpainting of the vacated region after the edited object moves out. However, current methods often either struggle to remove all traces of the object (e.g., object is not entirely removed in columns (d), (e), and (f) of Fig. 2 by the SOTA method DiffEditor (Mou et al. 2024a)), or hallucinate new unwanted artifacts in the vacated region. We attribute these issues to information leakage from similar objects through the SA mechanism (Dahary et al. 2024), and propose a leak-proof self-attention technique that prevents the attention to source object, target object, and similar objects in the image. Leak-proof SA leverages and controls the inter-region dependencies captured by SA to alleviate information leakage.

Intuitively, areas m_{old}, m_{new}, and m_{sim} all contain information about the to-be-edited object, and this information can be leaked to area m_{ipt} through the SA mechanism, where m_{old} is mask of to-be-edit object at the **source/old** location; m_{new} is m_{old} shifted to the the **target/new** location; m_{sim} is mask of other **similar** objects to the to-be-edited object (e.g., other apples in the multi-apple image in Fig. 1; see details on how to automatically obtain m_{sim} in the Appendix); and m_{ipt} equals the mask from $(m_{\text{old}} - m_{\text{new}})$ which represents the **to-be-inpainted** vacated region. To minimize the information leakage of the to-be-edited object and similar objects on the inpainted region, we strategically reset the corresponding elements (i.e., $m_{\text{old}} \cup m_{\text{new}} \cup m_{\text{sim}}$) in QK^T to a minimal value (i.e., $-\infty$). This strategy is activated for the target branch UNet call in all SA layers and at all time-steps to mitigate leakage and enable cohesive inpainting.

Editing Guidance with Latents Optimization

Mou et al. (2024b) propose a set of energy functions, which enforce feature correspondence to provide editing guidance. We utilize the same energy functions from DragonDiffusion (Mou et al. 2024b) to obtain additional editing guidance for object generation, harmonization, inpainting, and background consistency in our target branch.

Moreover, we aim to improve the efficiency of editing guidance. Mou et al. (2024a) showed that having a refinement loop that applies the editing guidance multiple times at a single time-step significantly enhances the performance. However, EG-based methods update the predicted noise ϵ while the loss function operates on the noisy latents z_t. To bridge this gap and propagate the guidance from ϵ to z_t, (Mou et al. 2024a) introduced "time travel" that requires a repetitive second round of DDIM inversion (Dhariwal and Nichol 2021). Therefore, the EG-based editing guidance can be computationally expensive in terms of NFEs.

[1]Injection refers to overwriting the respective attention K and V matrices with the previously saved ones.

Algorithm 1: Algorithm Overview of PixelMan

Require: VAE Encoder: $z_t = \mathcal{E}(I)$; VAE Decoder: $I = \mathcal{D}(z_t)$
Require: $\hat{\epsilon}_t, \{K, V\} = \text{UNet}(z_t, t)$
Require: $z_t = \text{FDP}(z_0, \epsilon)$; $\hat{z}_0 = \text{RGP}(z_t, \hat{\epsilon}_t, t)$
Require: $z_0^{\text{src}} = \mathcal{E}(I_{\text{src}})$; $z_0^{\text{man}} = \mathcal{E}(I_{\text{man}})$; $z_0^{\text{out}} = \mathcal{E}(I_{\text{man}})$
Require: source, target, and inpaint mask: m_{old}, m_{new}, m_{ipt}
for time-step $t \in \{T, T-1, ..., 1\}$ **do**
  $\epsilon \sim \mathcal{N}(0, I)$
  $z_t^{\text{src}} = \text{FDP}(z_0^{\text{src}}, \epsilon)$
  $z_t^{\text{man}} = \text{FDP}(z_0^{\text{man}}, \epsilon)$
  $z_t^{\text{tgt}} = \text{FDP}(z_0^{\text{out}}, \epsilon)$
  for repeat r **do** ▷ latents optimization
    $z_t^{\text{tgt}} \leftarrow z_t^{\text{tgt}} - \nabla_{z_t} \mathcal{E}(z_t^{\text{tgt}}, z_t^{\text{man}})$
  end for
  $\hat{\epsilon}_t^{\text{man}} = \text{UNet}(z_t^{\text{man}}, t)$
  $\hat{\epsilon}_t^{\text{src}}, \{K_{\text{src}}, V_{\text{src}}\} = \text{UNet}(z_t^{\text{src}}, t)$ ▷ save K,V
  $\hat{\epsilon}_t^{\text{tgt}} = \text{UNet}(z_t^{\text{tgt}}, t; \{K_{\text{src}}, V_{\text{src}}\})$ ▷ apply leak-proof SA
  $\hat{z}_0^{\text{man}} = \text{RGP}(z_t^{\text{man}}, \hat{\epsilon}_t^{\text{man}}, t)$
  $\hat{z}_0^{\text{tgt}} = \text{RGP}(z_t^{\text{tgt}}, \hat{\epsilon}_t^{\text{tgt}}, t)$
  if $t \in \{2, 1\}$ **then** ▷ i.e., last few time-steps
    $z_0^{\text{out}} = z_0^{\text{man}} + (\hat{z}_0^{\text{tgt}} - \hat{z}_0^{\text{man}})$ ▷ no masked-blending
  else
    $z_0^{\text{out}} = z_0^{\text{man}} + (\hat{z}_0^{\text{tgt}} - \hat{z}_0^{\text{man}}) \times (1 - m_{\text{new}})$ ▷ with mask
  end if
end for
Output: $I_{\text{out}} = \mathcal{D}(z_0^{\text{out}})$ ▷ the edited output image

Different from EG-based methods which updates the predicted noise ϵ, we propose a more efficient refinement strategy by applying the editing guidance directly to the target noisy latents z_t^{tgt} at each time-step t:

$$z_t^{\text{tgt}} \leftarrow z_t^{\text{tgt}} - \nabla_{z_t} \mathcal{E}(z_t^{\text{tgt}}, z_t^{\text{man}}). \tag{7}$$

The direct application of guidance at z_t^{tgt} eliminates the need for expensive tricks such as time travel. Our strategy is grounded in a solid theoretical foundation, as it leverages inference-time gradient descent optimization, which is also known as GSN (Chefer et al. 2023) in the text-to-image generation DM literatures. Furthermore, we demonstrate this efficiency through an ablation experiment in the Appendix.

In a refinement loop, we iteratively compute and apply the edit guidance to the target noisy latentsz_t^{tgt} as in Eq. (7). This iterative guidance process guarantees progressive deviation of z_t^{tgt} from z_t^{man}, resulting in the removal of the object from its old location and harmonization of the object with the context at its new location.

Experiments

First, we evaluate the effectiveness of PixelMan in the representative consistent object editing task, which is object repositioning. In addition, we also apply PixelMan on other consistent object editing tasks including object resizing, and object pasting to demonstrate the generalizability to different

Figure 2: **Visual comparisons on COCOEE dataset.** PixelMan achieves consistent object editing for object repositioning with lower latency and fewer inference steps, while better preserving image consistency and achieving cohesive inpainting.

tasks (see Appendix). For the object repositioning task, we perform extensive quantitative evaluation and visual comparisons, both against the existing methods, as well as ablation studies on the various components of PixelMan (placed in the Appendix due to space constraints). In the Appendix, we also provide complete details of all the quantitative results and more visual comparisons.

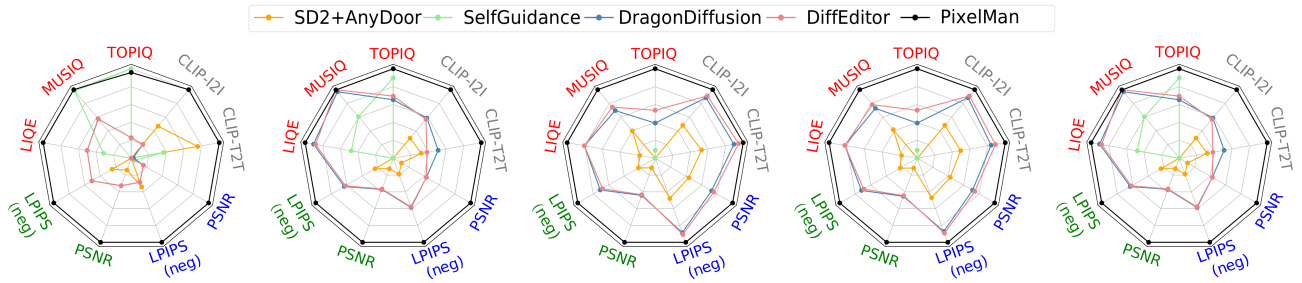
For our experiments, we adopted subsets of two challenging datasets, namely COCOEE (Lin et al. 2014; Yang et al. 2022) and ReS (Wang et al. 2024) (detailed in the Appendix). To comprehensively evaluate performance quantitatively, we adopted 9 metrics from 4 categories (elaborated in the Appendix): *Image Quality Assessment* (3), *Object Consistency* (2), *Background Consistency* (2), and *Semantic Consistency* (2). For *Efficiency*, we compare the number of inference steps, NFEs (i.e., number of UNet calls), and the average latency over 10 runs.

Results on Object Repositioning

We compare PixelMan with three existing object repositioning methods including, SelfGuidance (Epstein et al. 2023), DragonDiffusion (Mou et al. 2024b), and DiffEditor (Mou et al. 2024a) (SOTA). All training-free methods are evaluated based on SDv1.5 (Rombach et al. 2022; AI 2022a) to align with (Mou et al. 2024b,a). For thorough evaluations, we also consider a training-based baseline using SDv2-Inpainting Model (Rombach et al. 2022; AI 2022b) to inpaint the original location and then use AnyDoor (Chen et al. 2024) for inserting the object at the target location.

Overall performance. In Figs. 3a, 3b, and 3c, we compare PixelMan against the four contenders on the COCOEE dataset at the same number of inference steps (8, 16, and 50 steps respectively). At 50 steps, PixelMan outperforms all other methods in 9 out of 9 metrics. At 16 steps, PixelMan outperforms other methods in 8 out of 9 metrics, while being second place on the MUSIQ IQA metric. At 8 steps, PixelMan scores the best in 8 out of 9 metrics, while being second place on the TOPIQ IQA metric. Overall, PixelMan outperforms the other methods at the same number of steps. In the Appendix, we provide the full quantitative results and visual comparisons of all methods at both 16 and 50 steps.

Efficiency. More importantly, PixelMan achieves superior performance with fewer NFEs than existing methods. We attribute this to our three-branched inversion-free sampling approach that avoids quality degradation at 16 steps, seen in methods (Epstein et al. 2023; Mou et al. 2024b,a) that rely on DDIM inversion (e.g., row "DiffEditor 16 steps" of Fig. 2). As shown in Table 1, PixelMan at 16 steps requires 112 fewer computations and is 15 seconds faster than the SOTA DiffEditor on COCOEE. Despite being faster,



(a) COCOEE dataset, all methods using 8 steps

(b) COCOEE dataset, all methods using 16 steps

(c) COCOEE dataset, all methods using 50 steps

(d) COCOEE dataset, PixelMan using 16 steps, others using 50 steps

(e) ReS dataset, PixelMan using 16 steps, others using 50 steps

Figure 3: **Radar charts** that shows *normalized* evaluation metric values of different methods. TOPIQ, MUSIQ, LIQE belong to IQA; LPIPS (neg) and PSNR belong to Object Consistency; LPIPS (neg) and PSNR belong to Background Consistency; and CLIP-T2T and CLIP-I2I belong to Semantic Consistency. **Detailed results and additional comparisons in Appendix.**

	#Steps	NFEs	COCOEE avg(lat.)	ReS avg(lat.)
SD2+AnyDoor	50	100	15	16
SelfGuidance	50	100	11	14
DragonDiffusion	50	160	23	30
DiffEditor	50	176	24	32
PixelMan (ours)	16	64	9	11

Table 1: **Efficiency comparisons.** PixelMan at 16 steps performs 112 fewer NFEs and is 15 seconds faster than DiffEditor (Mou et al. 2024a) on the COCOEE dataset.

PixelMan's quality at 16 steps surpasses DiffEditor's at 50 steps (additional examples in the Appendix). Therefore, hereafter, we directly compare PixelMan at 16 steps to other methods at 50 steps in the following evaluation categories.

Image quality. In Fig. 3d, PixelMan (16 steps) achieves significantly better image quality in all three IQA metrics than the other methods (50 steps) on COCOEE. In Fig. 3e, PixelMan has similar image quality to DragonDiffusion and DiffEditor on ReS dataset even when using significantly fewer steps. In visual comparisons, we observe PixelMan achieves overall better image quality than other methods while being more efficient. This includes less artifacts, more natural colors, well-blended objects and backgrounds, and natural lighting and shadow.

Object consistency. PixelMan (16 steps) excels in object consistency on both COCOEE and ReS datasets (Figs. 3d, 3e), as measured by LPIPS (neg) and PSNR. Our three-branched inversion-free sampling approach helps the faithful reproduction of the object at the new location since we always anchor the output latents to the pixel-manipulated latents which ensures the moved object to be consistent with the original object. This is evident in Fig. 2 and Fig. 6 in Appendix, where PixelMan consistently preserves details like shape, color, and texture (e.g., clock, bird, airplane, orange).

Background consistency. On both COCOEE and ReS datasets (i.e., Fig. 3d and Fig. 3e), PixelMan outperforms all other methods in both background consistency metrics LPIPS (neg) and PSNR. In the visual examples in Fig. 2, we observe the background in PixelMan's edited images are more consistent with the source image (e.g., the grass texture and color in (b) and the water color in (e)).

Inpainting. We provide abundant visual comparisons to assess the inpainting quality in Fig. 2 and in Figs. 6, 7, 8, 9, and 10 (in the Appendix). Here, we see PixelMan excels at removing objects (e.g., plane, pillow, orange in Fig. 2) while preserving the surrounding scene. Conversely, other methods either leave traces of the original object or introduce new artifacts in the inpainted area.

Semantic consistency. PixelMan outperforms all methods on COCOEE and is best in CLIP-I2I on ReS and remains competitive in CLIP-T2T (see Fig. 3). PixelMan preserves the original semantics of the source image, while maintaining consistency in object, background and better inpainting quality (e.g., 2 instead of 3 oranges in Fig. 2 (h)).

Conclusion

We propose PixelMan, an inversion-free and training-free method for achieving consistent object editing via Pixel Manipulation and generation. PixelMan maintains image consistency by directly creating a duplicate copy of the source object at target location in the pixel space, and we introduce an efficient sampling approach to iteratively harmonize the manipulated object into the target location and inpaint its original location. The key to image consistency is anchoring the output image to be generated to the pixel-manipulated image and introducing various consistency-preserving optimization techniques during inference. Moreover, we propose a leak-proof SA technique to enable cohesive inpainting by addressing the attention leakage issue which is a root cause of failed inpainting. Quantitative results on the COCOEE and ReS datasets and extensive visual comparisons show that PixelMan achieves superior performance in consistency metrics for object, background, and image semantics while achieving higher or comparable performance in IQA metrics. As a training-free method, PixelMan only requires 16 inference steps with lower latency and a lower number of NFEs than current popular methods.

References

Adobe. 2023. AI Photo Editor: Edit Images with AI in Photoshop - Adobe. https://www.adobe.com/products/photoshop/ai.html. Accessed: 2024-05-22.

AI, S. 2022a. SDv1.5. https://huggingface.co/runwayml/stable-diffusion-v1-5. Accessed: 2024-05-14.

AI, S. 2022b. SDv2-inpainting. https://huggingface.co/stabilityai/stable-diffusion-2-inpainting. Accessed: 2024-05-14.

AI, S. 2023. SDXL-Turbo. https://huggingface.co/stabilityai/sdxl-turbo. Accessed: 2024-05-14.

Alaluf, Y.; Garibi, D.; Patashnik, O.; Averbuch-Elor, H.; and Cohen-Or, D. 2023. Cross-image attention for zero-shot appearance transfer. *arXiv preprint arXiv:2311.03335*.

Brooks, T.; Holynski, A.; and Efros, A. A. 2023. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18392–18402.

Cao, M.; Wang, X.; Qi, Z.; Shan, Y.; Qie, X.; and Zheng, Y. 2023. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 22560–22570.

Chang, H.; Zhang, H.; Barber, J.; Maschinot, A.; Lezama, J.; Jiang, L.; Yang, M.-H.; Murphy, K.; Freeman, W. T.; Rubinstein, M.; et al. 2023. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*.

Chefer, H.; Alaluf, Y.; Vinker, Y.; Wolf, L.; and Cohen-Or, D. 2023. Attend-and-Excite: Attention-Based Semantic Guidance for Text-to-Image Diffusion Models. arXiv:2301.13826.

Chen, X.; Huang, L.; Liu, Y.; Shen, Y.; Zhao, D.; and Zhao, H. 2024. Anydoor: Zero-shot object-level image customization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6593–6602.

Dahary, O.; Patashnik, O.; Aberman, K.; and Cohen-Or, D. 2024. Be Yourself: Bounded Attention for Multi-Subject Text-to-Image Generation. *arXiv preprint arXiv:2403.16990*.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34: 8780–8794.

Duan, X.; Cui, S.; Kang, G.; Zhang, B.; Fei, Z.; Fan, M.; and Huang, J. 2024. Tuning-free inversion-enhanced control for consistent image editing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 1644–1652.

Endo, Y. 2022. User-Controllable Latent Transformer for StyleGAN Image Layout Editing. *Computer Graphics Forum*, 41(7): 395–406.

Epstein, D.; Jabri, A.; Poole, B.; Efros, A.; and Holynski, A. 2023. Diffusion self-guidance for controllable image generation. *Advances in Neural Information Processing Systems*, 36: 16222–16239.

Garibi, D.; Patashnik, O.; Voynov, A.; Averbuch-Elor, H.; and Cohen-Or, D. 2024. ReNoise: Real Image Inversion Through Iterative Noising. *arXiv preprint arXiv:2403.14602*.

Google. 2023. Google Photos MagicEditor. https://blog.google/products/photos/google-photos-magic-editor-pixel-io-2023/. Accessed: 2024-05-22.

Hertz, A.; Mokady, R.; Tenenbaum, J.; Aberman, K.; Pritch, Y.; and Cohen-Or, D. 2022. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*.

Hertz, A.; Voynov, A.; Fruchter, S.; and Cohen-Or, D. 2023. Style aligned image generation via shared attention. *arXiv preprint arXiv:2312.02133*.

Jeong, J.; Kim, J.; Choi, Y.; Lee, G.; and Uh, Y. 2024. Visual Style Prompting with Swapping Self-Attention. *arXiv preprint arXiv:2402.12974*.

Kawar, B.; Zada, S.; Lang, O.; Tov, O.; Chang, H.; Dekel, T.; Mosseri, I.; and Irani, M. 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.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 740–755. Springer.

Luo, S.; Tan, Y.; Huang, L.; Li, J.; and Zhao, H. 2023. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*.

Mou, C.; Wang, X.; Song, J.; Shan, Y.; and Zhang, J. 2024a. DiffEditor: Boosting Accuracy and Flexibility on Diffusion-based Image Editing. *arXiv preprint arXiv:2402.02583*.

Mou, C.; Wang, X.; Song, J.; Shan, Y.; and Zhang, J. 2024b. DragonDiffusion: Enabling Drag-style Manipulation on Diffusion Models. In *The Twelfth International Conference on Learning Representations*.

Pan, X.; Tewari, A.; Leimkühler, T.; Liu, L.; Meka, A.; and Theobalt, C. 2023. Drag your gan: Interactive point-based manipulation on the generative image manifold. In *ACM SIGGRAPH 2023 Conference Proceedings*, 1–11.

Parmar, G.; Kumar Singh, K.; Zhang, R.; Li, Y.; Lu, J.; and Zhu, J.-Y. 2023. Zero-shot image-to-image translation. In *ACM SIGGRAPH 2023 Conference Proceedings*, 1–11.

Patashnik, O.; Garibi, D.; Azuri, I.; Averbuch-Elor, H.; and Cohen-Or, D. 2023. Localizing object-level shape variations with text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 23051–23061.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Saharia, C.; Chan, W.; Chang, H.; Lee, C.; Ho, J.; Salimans, T.; Fleet, D.; and Norouzi, M. 2022a. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 conference proceedings*, 1–10.

Saharia, C.; Chan, W.; Saxena, S.; Li, L.; Whang, J.; Denton, E. L.; Ghasemipour, K.; Gontijo Lopes, R.; Karagol Ayan, B.; Salimans, T.; et al. 2022b. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35: 36479–36494.

Shi, Y.; Xue, C.; Pan, J.; Zhang, W.; Tan, V. Y.; and Bai, S. 2023. Dragdiffusion: Harnessing diffusion models for interactive point-based image editing. *arXiv preprint arXiv:2306.14435*.

Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2020. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*.

Tumanyan, N.; Geyer, M.; Bagon, S.; and Dekel, T. 2023. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1921–1930.

Wang, Y.; Cao, C.; Dong, Q.; Li, Y.; and Fu, Y. 2024. Repositioning the Subject within Image. *arXiv preprint arXiv:2401.16861*.

Winter, D.; Cohen, M.; Fruchter, S.; Pritch, Y.; Rav-Acha, A.; and Hoshen, Y. 2024. ObjectDrop: Bootstrapping Counterfactuals for Photorealistic Object Removal and Insertion. *arXiv preprint arXiv:2403.18818*.

Xu, S.; Huang, Y.; Pan, J.; Ma, Z.; and Chai, J. 2024. Inversion-Free Image Editing with Natural Language. In *Conference on Computer Vision and Pattern Recognition 2024*.

Yang, B.; Gu, S.; Zhang, B.; Zhang, T.; Chen, X.; Sun, X.; Chen, D.; and Wen, F. 2022. Paint by Example: Exemplar-based Image Editing with Diffusion Models. *arXiv preprint arXiv:2211.13227*.

Zhang, L.; Rao, A.; and Agrawala, M. 2023. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3836–3847.