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]Dept. ECE, University of Alberta
[2]Huawei Technologies Canada
[3]Huawei Kirin Solution

PixelMan@AAAI2025
liyaojiang1998.github.io

UNIVERSITY OF ALBERTA

HUAWEI

AAAI-25 / IAAI-25 / EAAI-25
FEBRUARY 25 – MARCH 4, 2025 | PHILADELPHIA, USA

Project Page: https://liyaojiang1998.github.io/projects/PixelMan/

Background - Image Editing

Diffusion models enable powerful AI image editing applications

Promising results on **text-guided rigid** image editing

- Changing the color, texture, attributes, and style

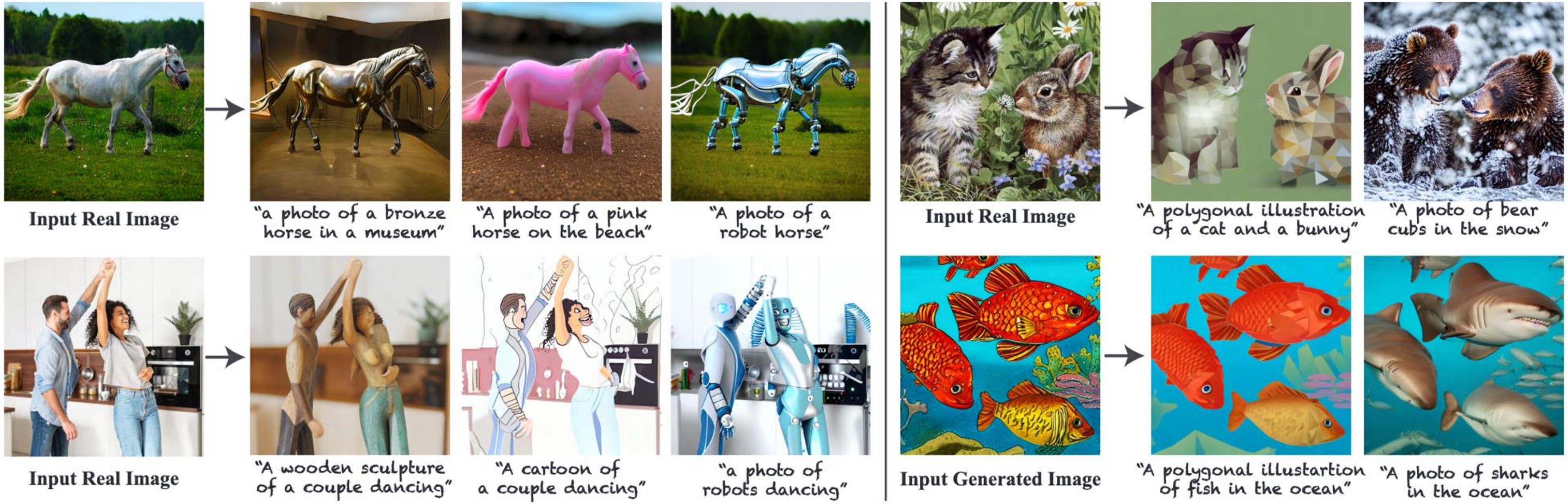


Figure: Text-guided rigid image editing, from Plug-and-Play (Tumanyan et al., CVPR 2023).

Background - Consistent Object Editing

Consistent object editing

- Preserve the **consistency** of object/background, without changing color/texture
- Only edit certain **non-rigid** object attributes (e.g., position, size, composition)
- Typical tasks: object repositioning, resizing, pasting

A challenging task involving multiple sub-tasks

1. Faithful reproduction of source object at the target location
2. Maintain background scene details
3. Harmonization of the new object into its surrounding context
4. Inpainting the vacated area with cohesive background

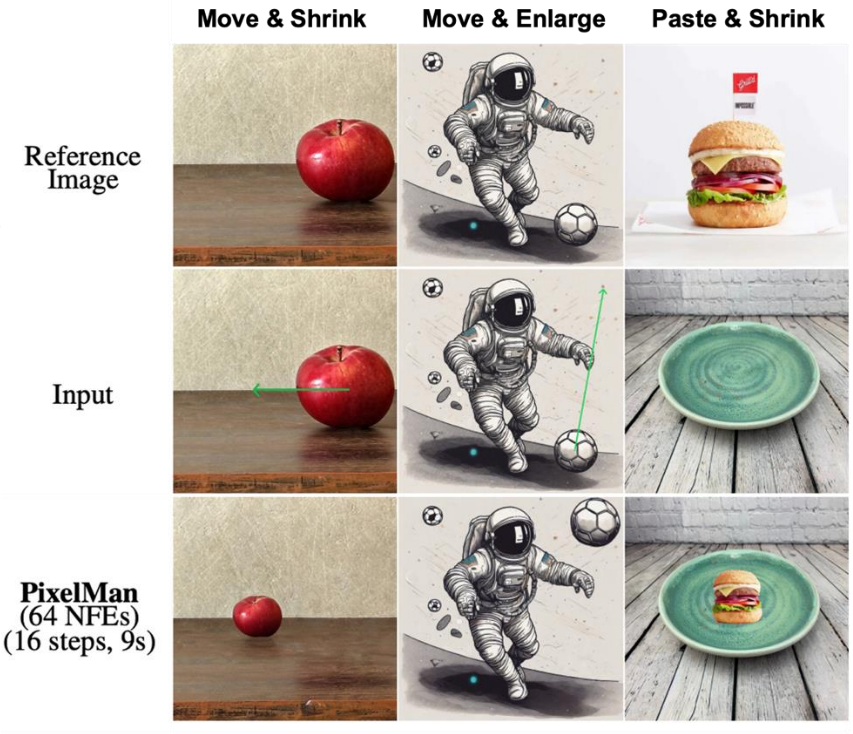


Figure: Typical consistent object editing tasks.

Challenges in Consistent Object Editing

Issues in existing methods

1. Low efficiency
   - Rely on DDIM Inversion to reconstruct original image, which requires many (e.g., at least 50) steps, compromising quality when reducing # steps
2. Low object and background consistency
   - Altered object identity, inconsistent background
3. Incomplete & incoherent inpainting
   - Fail to inpaint vacated area with cohesive background

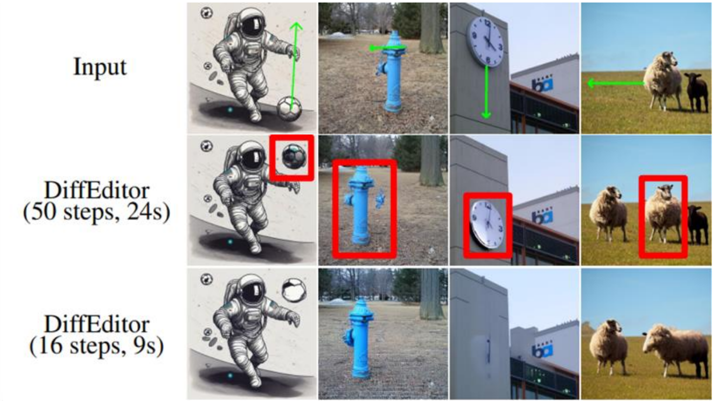


Figure: Issues faced by existing methods.

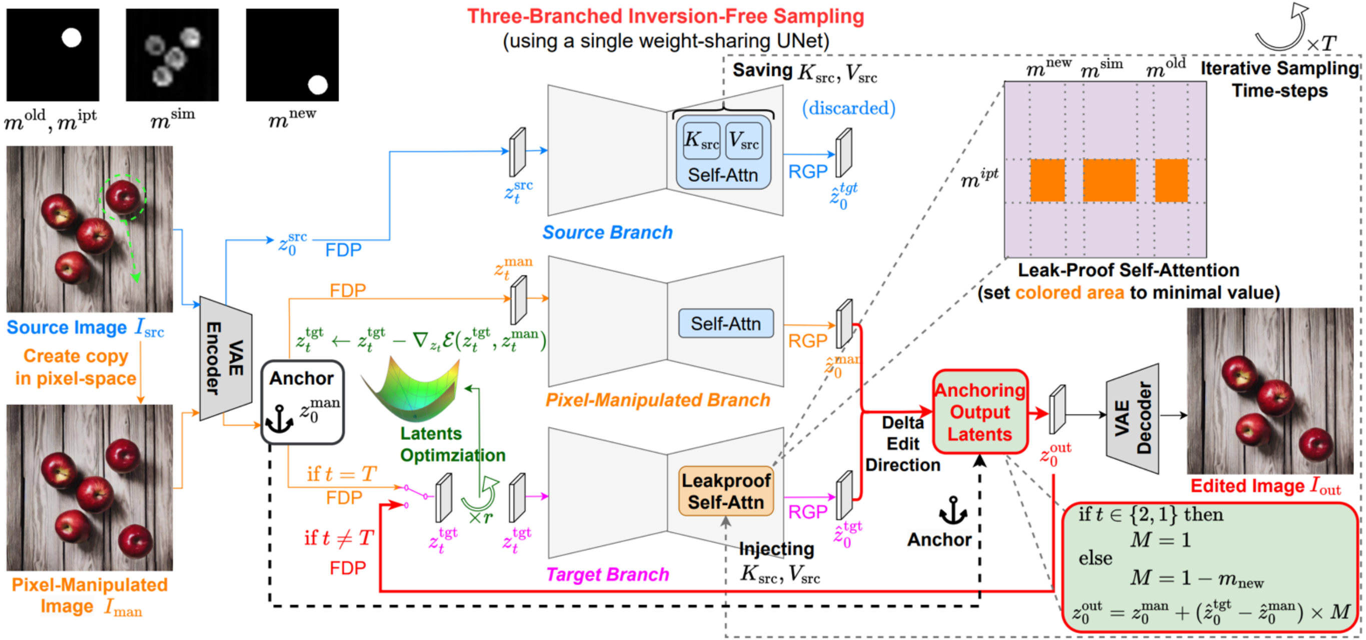
Baselines

- AnyDoor (CVPR 2024, training-based)
  - Collect task-specific (i.e. object pasting) dataset and need costly training of the DM
- SelfGuidance (NeurIPS 2023, training-free)
  - At inference, update the predicted noise with energy functions defined on CA maps
  - Rely on inefficient DDIM inversion, struggles to produce a consistent reconstruction
- DragonDiffusion (ICLR 2024, training-free)
  - Define energy functions to minimize feature similarity between source and target object/background; Also rely on DDIM Inversion
- DiffEditor (CVPR 2024, training-free)
  - Improving consistency with regional SDE sampling and score-based gradient guidance

Our Method - Pixel Manipulation and Generation (PixelMan)

Our Techniques

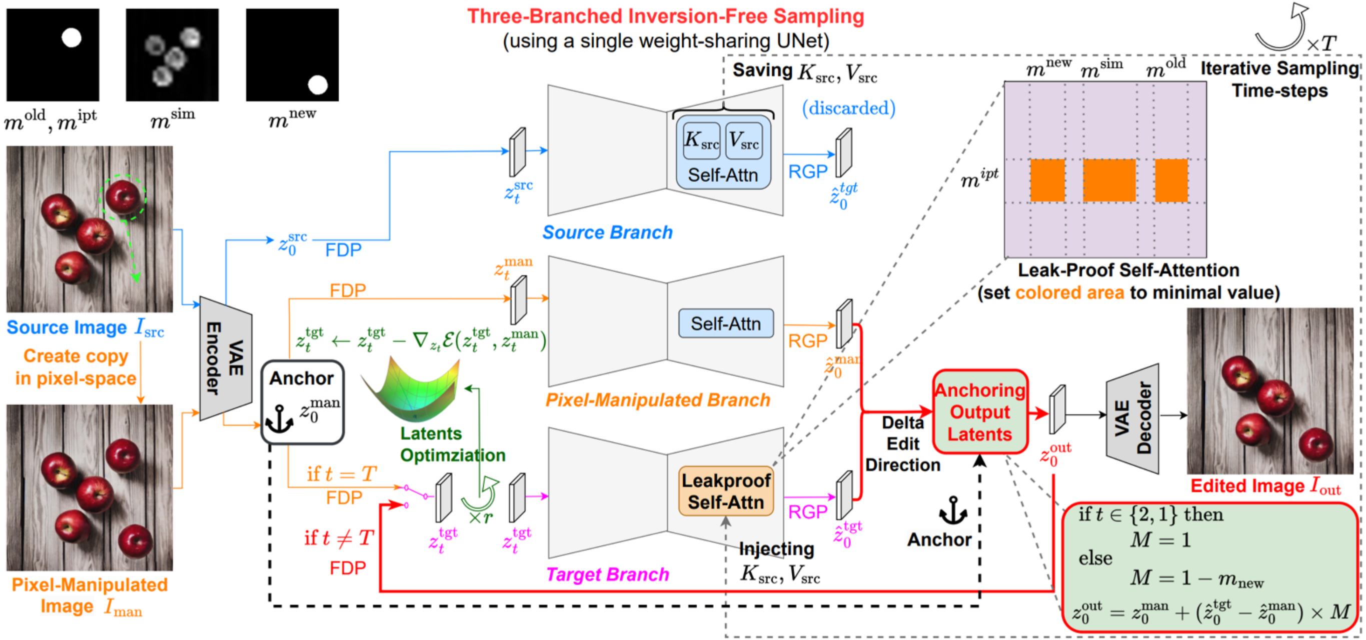
1. Three-branched inversion-free sampling
   - To improve efficiency, and to preserve consistency in object and background



Our Method - Pixel Manipulation and Generation (PixelMan)

Our Techniques

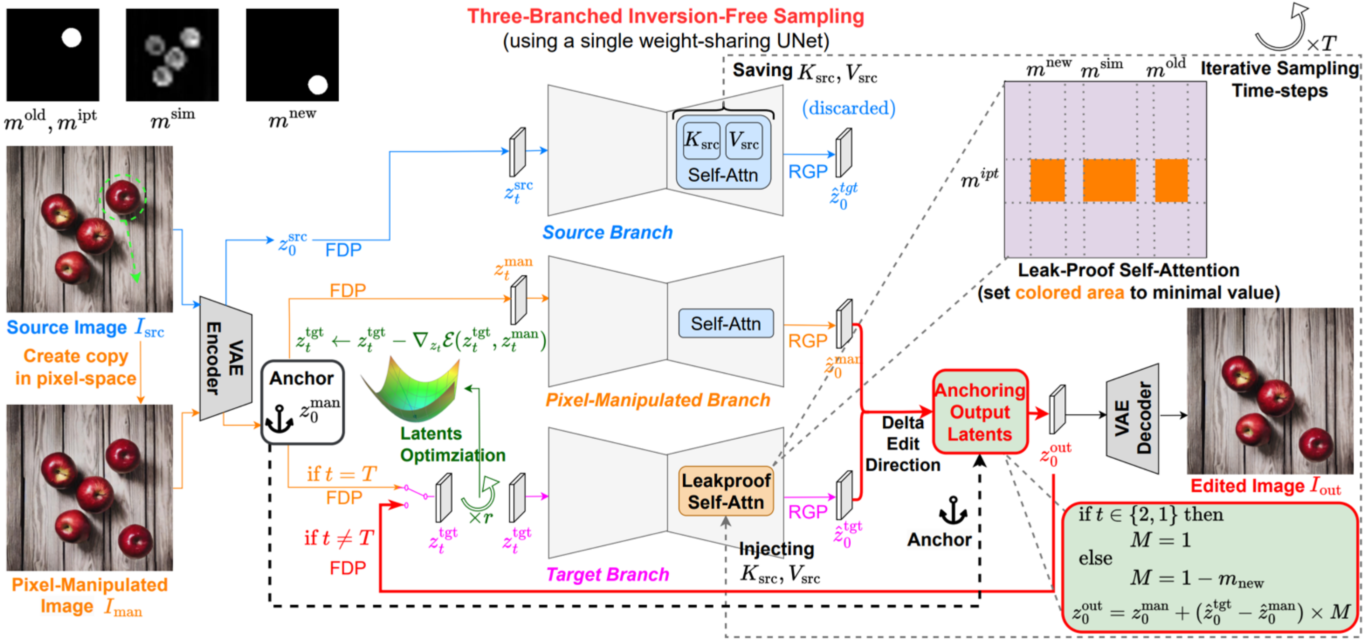
1. Three-branched inversion-free sampling
   - To improve efficiency, and to preserve consistency in object and background
2. Editing guidance techniques
   - To generate the inpainting and harmonization edits



Our Method - Pixel Manipulation and Generation (PixelMan)

Our Techniques

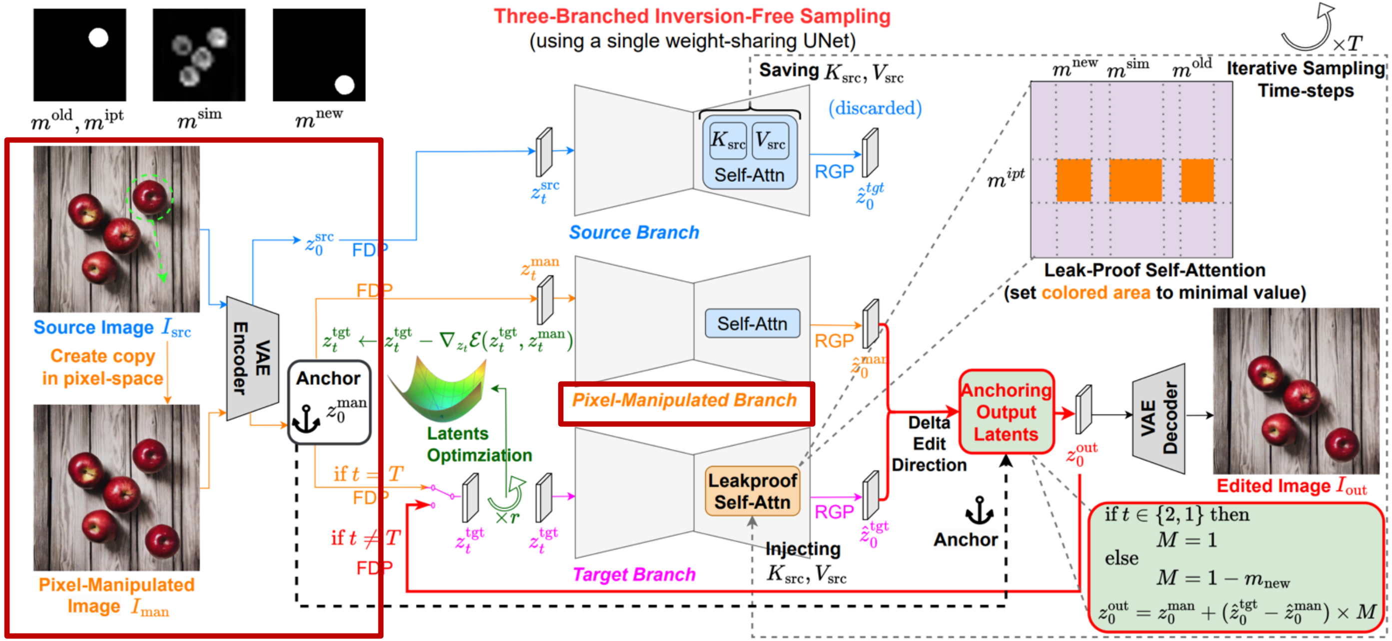
1. Three-branched inversion-free sampling
   - To improve efficiency, and to preserve consistency in object and background
2. Editing guidance techniques
   - To generate the inpainting and harmonization edits
3. Leak-proof self-attention
   - To achieve complete and cohesive inpainting



Our Method - Pixel Manipulation and Generation (PixelMan)

1. Three-branched inversion-free sampling

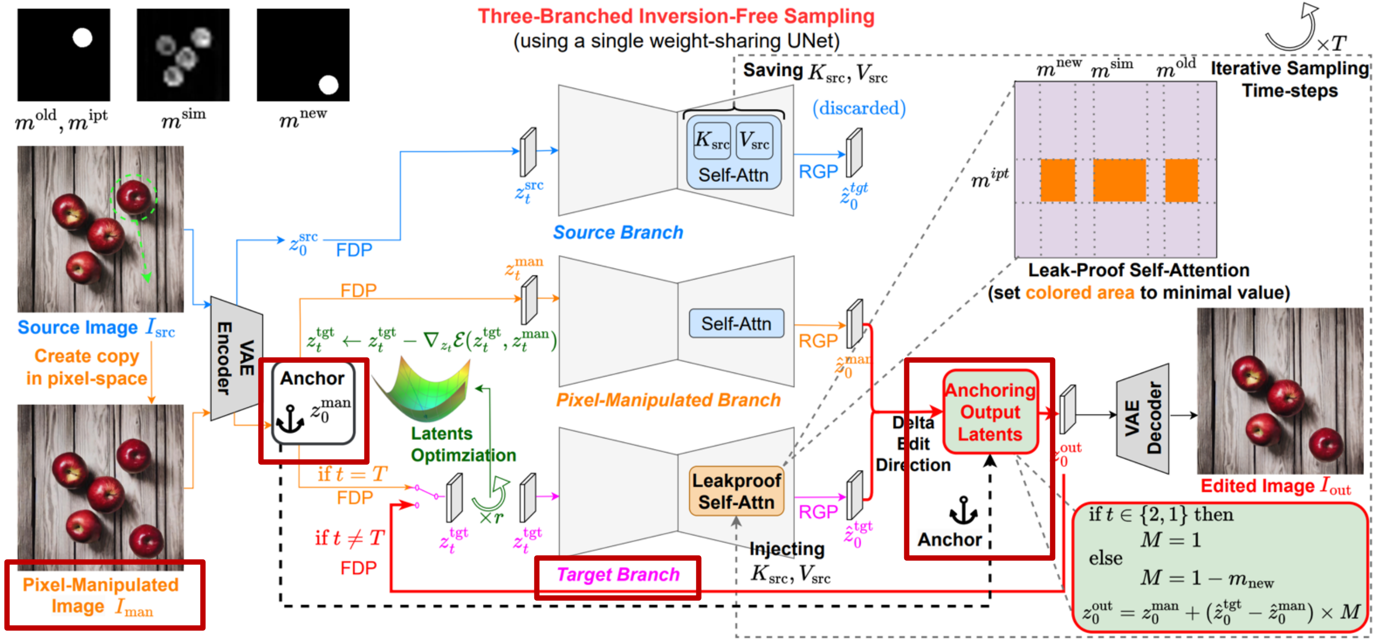
- **Pixel Manipulation:** reproduce the object and background with high consistency, while being inversion-free
  - Pixel-manipulated branch: copy the source object to target location in pixel space



Our Method - Pixel Manipulation and Generation (PixelMan)

1. Three-branched inversion-free sampling

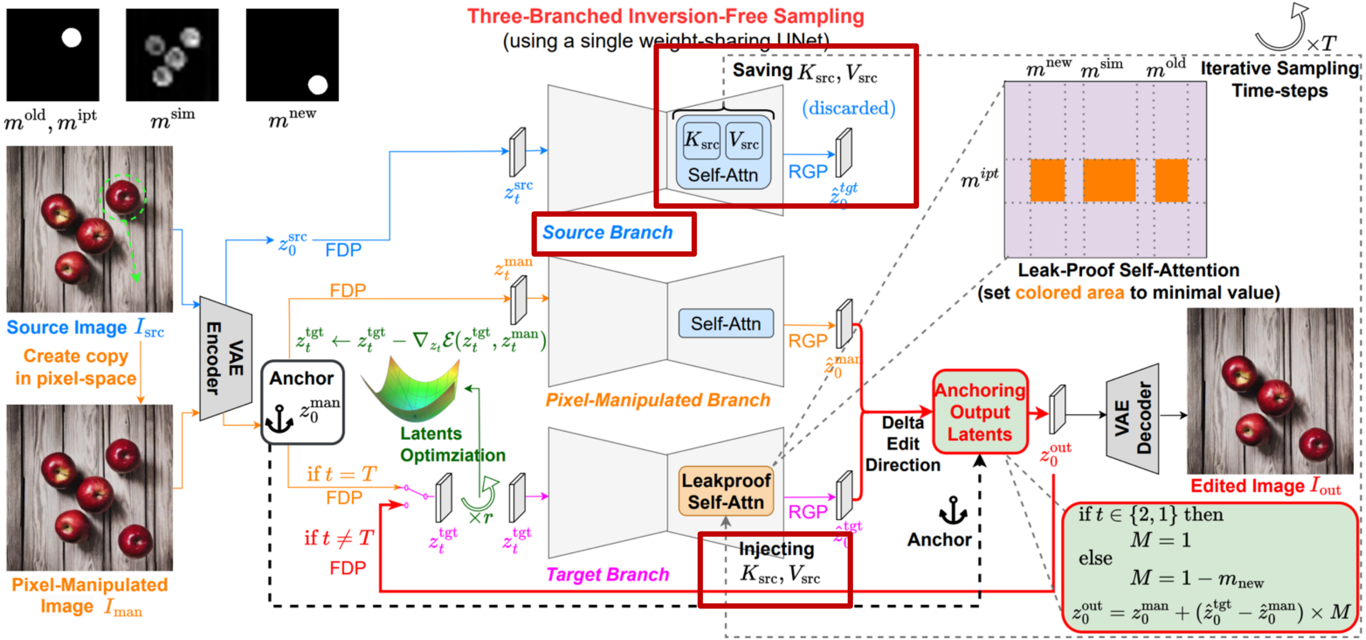
- **Pixel Manipulation:** reproduce the object and background with high consistency, while being inversion-free
  - Pixel-manipulated branch: copy the source object to target location in pixel space
  - Target branch: at each step, always **anchor** the target latents to the pixel-manipulated latents



Our Method - Pixel Manipulation and Generation (PixelMan)

1. Three-branched inversion-free sampling

- **Pixel Manipulation:** reproduce the object and background with high consistency, while being inversion-free
  - Pixel-manipulated branch: copy the source object to target location in pixel space
  - Target branch: at each step, always **anchor** the target latents to the pixel-manipulated latents
  - Source branch: preserve uncontaminated K, V features as context for generating harmonization effects (e.g. lighting, shadow, edge blending)



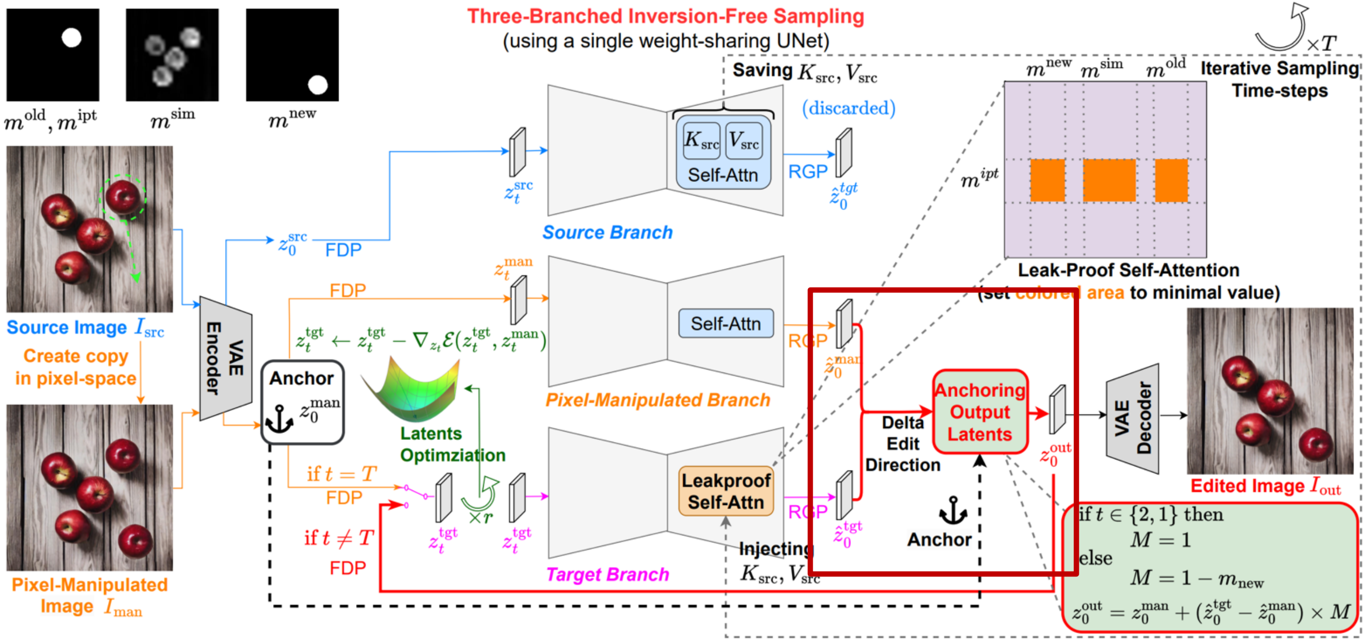
Our Method - Pixel Manipulation and Generation (PixelMan)

2. Editing guidance techniques

- Output Latents = Anchor + (Predicted Target Latents – Predicted Pixel-Manipulated Latents) x Blending Mask

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

- **Generation:** find the delta editing direction to be added on top of the anchor (i.e., generate harmonization and inpainting)



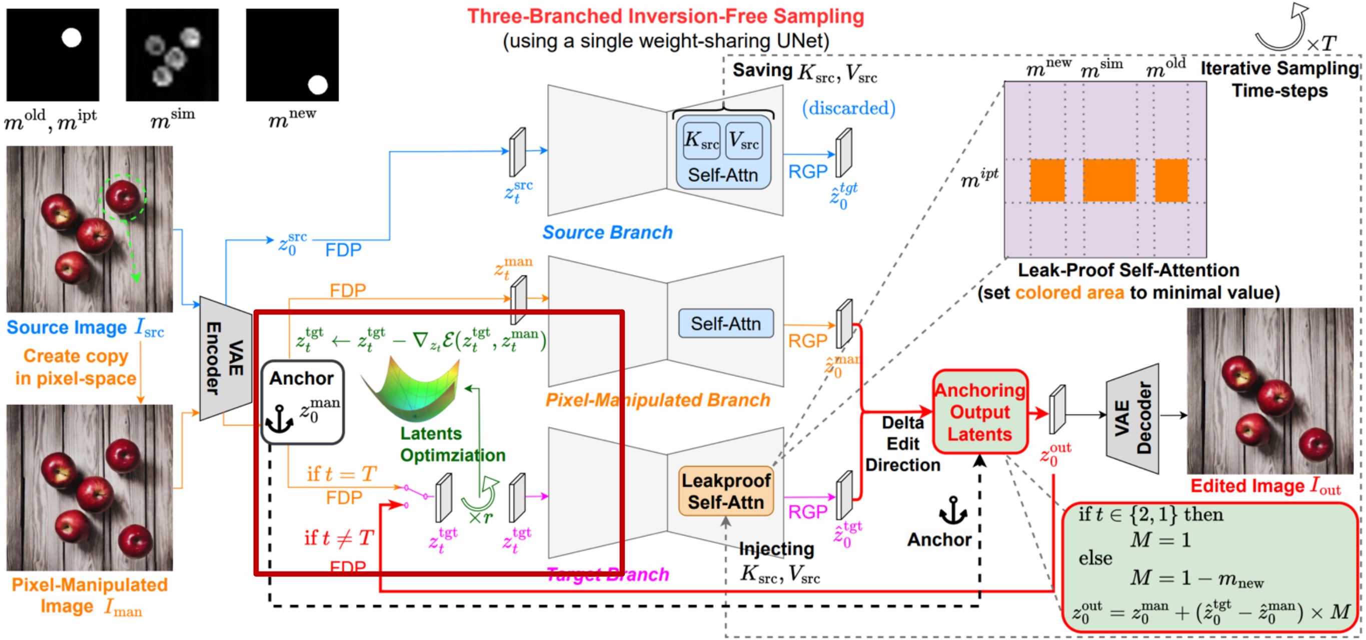
Our Method - Pixel Manipulation and Generation (PixelMan)

2. Editing guidance techniques

- Output Latents = Anchor + (Predicted Target Latents – Predicted Pixel-Manipulated Latents) x Blending Mask

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

- **Generation:** find the delta editing direction to be added on top of the anchor (i.e., generate harmonization and inpainting)
    - Editing guidance based on energy functions with latents optimization (update z instead of ϵ, reduces #NFE)



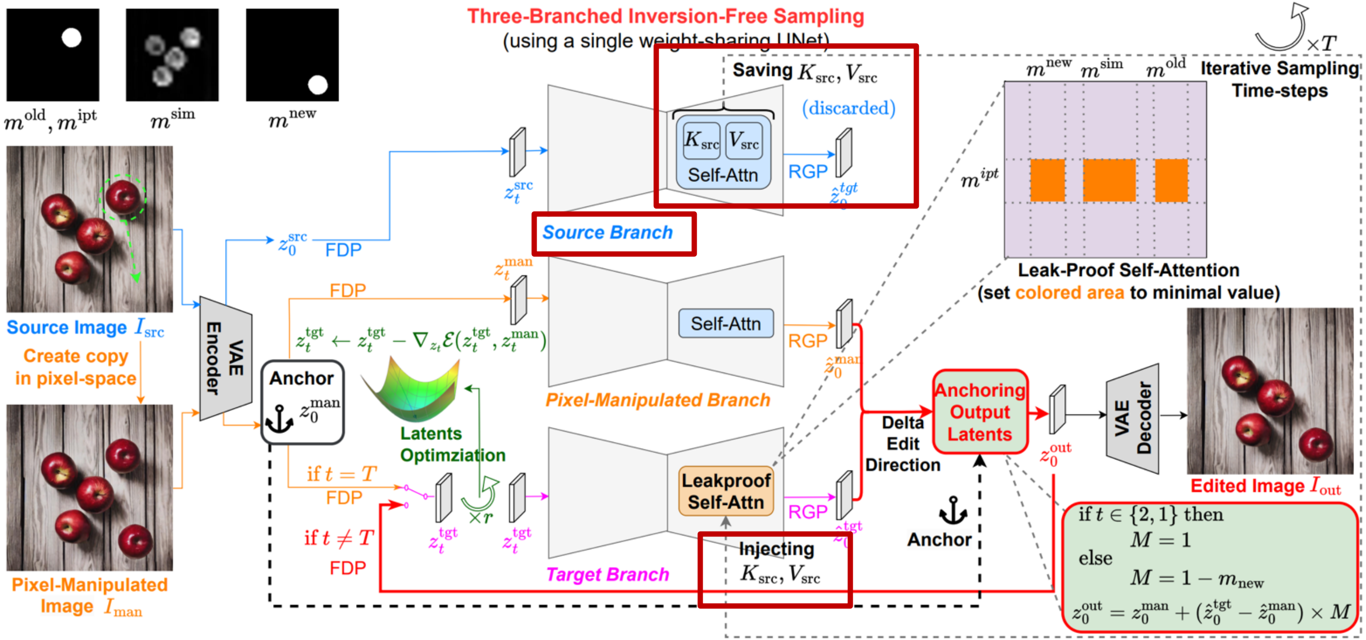
Our Method - Pixel Manipulation and Generation (PixelMan)

2. Editing guidance techniques

- Output Latents = Anchor + (Predicted Target Latents – Predicted Pixel-Manipulated Latents) x Blending Mask

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

- **Generation:** find the delta editing direction to be added on top of the anchor (i.e., generate harmonization and inpainting)
  - Editing guidance based on energy functions with latents optimization (update z instead of ϵ, reduces #NFE)
  - Injection of source K, V features into the target branch



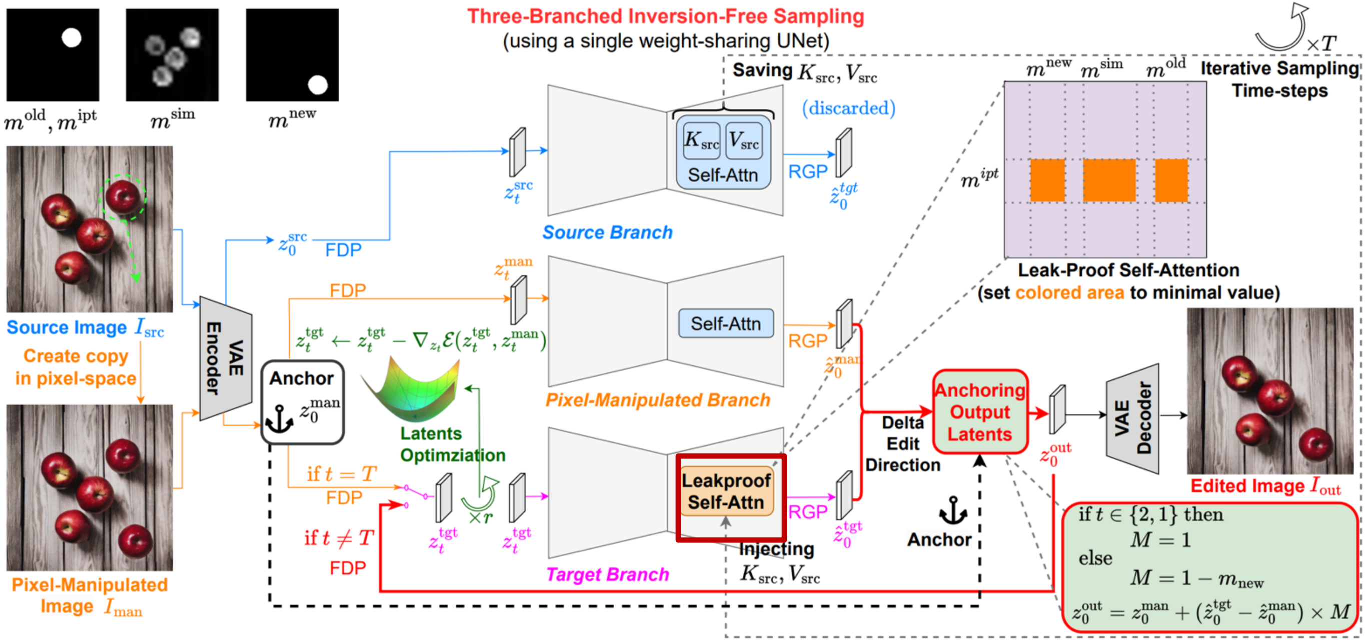
Our Method - Pixel Manipulation and Generation (PixelMan)

2. Editing guidance techniques

- Output Latents = Anchor + (Predicted Target Latents – Predicted Pixel-Manipulated Latents) x Blending Mask

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

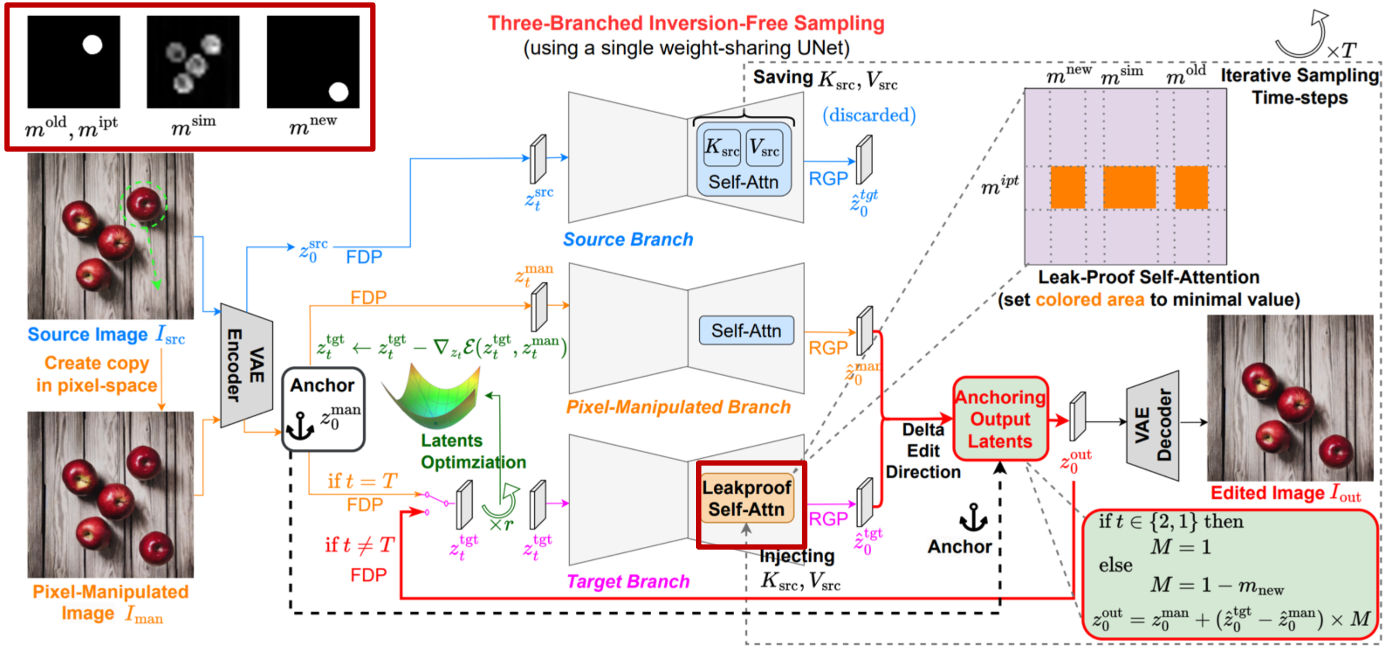
- **Generation:** find the delta editing direction to be added on top of the anchor (i.e., generate harmonization and inpainting)
  - Editing guidance based on energy functions with latents optimization (update z instead of ϵ, reduces #NFE)
  - Injection of source K, V features into the target branch
  - Apply leak-proof self-attention in target branch



Our Method - Pixel Manipulation and Generation (PixelMan)

3. Leak-proof self-attention

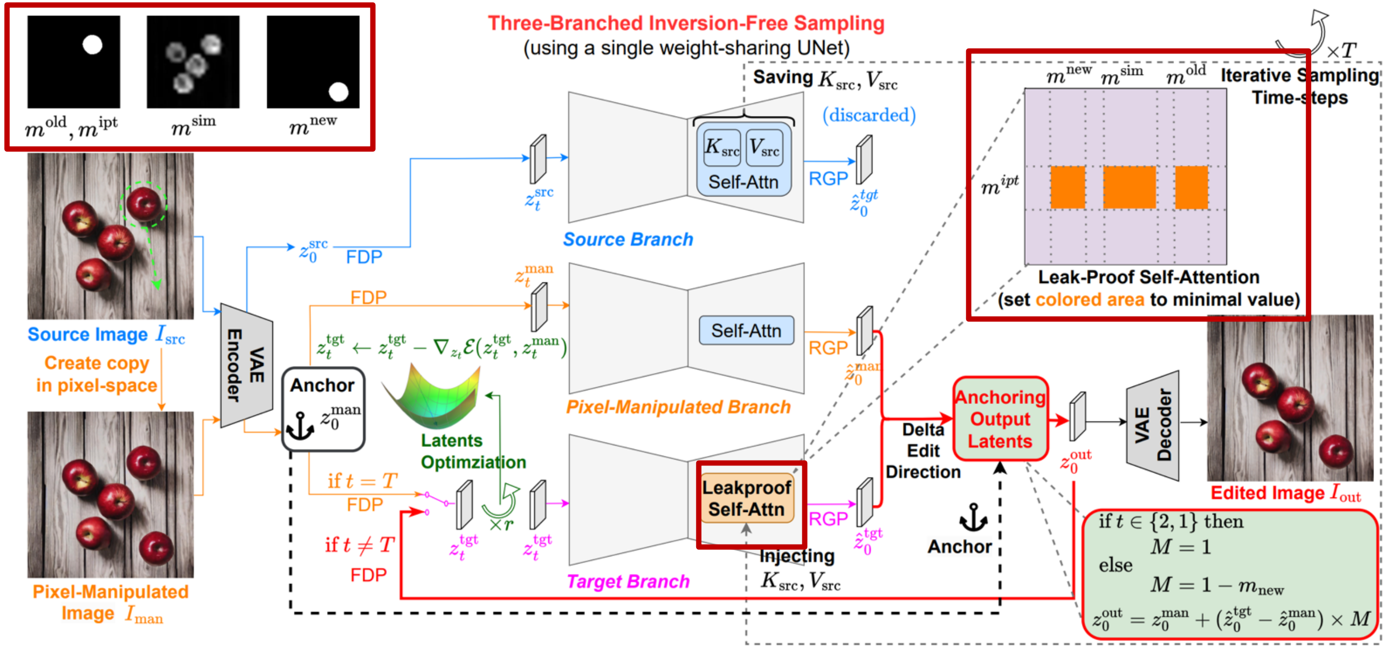
- Root cause of inpainting failure
  - Information leakage from similar objects through the SA



Our Method - Pixel Manipulation and Generation (PixelMan)

3. Leak-proof self-attention

- Root cause of inpainting failure
  - Information leakage from similar objects through the SA
- Solution: prevent attention to source, target, and similar objects
  - Set the corresponding QK^T elements to minimal values
- Achieve complete and cohesive inpainting



Evaluation - Datasets

Object Repositioning Datasets

Each sample contains: image, object mask, diff vector

- **COCOEE dataset:** we manually annotate 100 samples for object repositioning
  - Sampled from the COCOEE dataset by Yang et al. (2022) (a subset of MSCOCO)
  - Annotator use Segment Anything Model to pick an object
  - Pick the start and end point for moving (i.e., diff vector)
- **ReS dataset:** open-source dataset by Wang et al. (2024)
  - Manually created in real-world (i.e., physically move one object)
  - Before and after images captured with phone cameras
  - 162 samples, excluding occlusion cases (i.e., behind other objects after moving)
  - A challenging dataset due to changes in scale of the moved objects, lighting, shadows, etc.

Image | Object Mask | Diff Vector

[[308, 299], [88, 299]]

Figure: Data sample from COCOEE dataset.

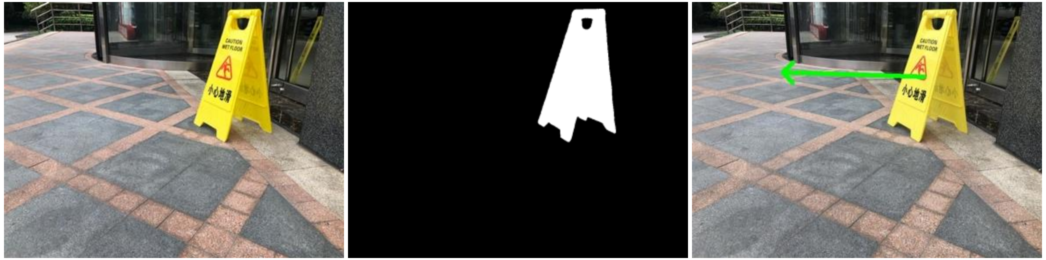


Image | Object Mask | Diff Vector

[[467, 149], [182, 140]]

Figure: Data sample from ReS dataset.

Evaluation - Metrics

Efficiency:

- Latency (in seconds), NFEs (number of function evaluations, i.e., UNet calls)

Image Quality Assessment (IQA): TOPIQ, MUSIQ, LIQE

- Overall perceptual visual quality

Evaluating the consistency (for the object, background, and semantic) before and after editing

Object Consistency: LPIPS, PSNR (Note: LPIPS is smaller the better)

- Similarity of the moved object to the original object

Background Consistency: LPIPS, PSNR

- Similarity of the background in the edited image to the background in the original image

Semantic Consistency:

- CLIP-I2I: CLIP Score (source image, edited image)
    - Similarity between the semantics of the source image and the edited image
- CLIP-T2T: CLIP Score (source caption, edited caption)
    - Captions are generated with BLIP captioning model
    - e.g., “a seagull flying over a body of water”

Original Image

Original Background

Original Object (Pixel Manipulated)

Evaluation – Editing Quality and Efficiency

PixelMan improves editing quality

- Object is consistent to the source (attributes and identity)
- Background is preserved after editing (texture and color)
- Original object is completed removed and inpainted with cohesively background

- PixelMan (@ 16 steps) has better quality than competitive training-free and training-based methods (@ 50 steps)
- **While having better efficiency**
    - Reduce latency: 24s -> 9s
    - Reduce #NFEs: 176 -> 64

	#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

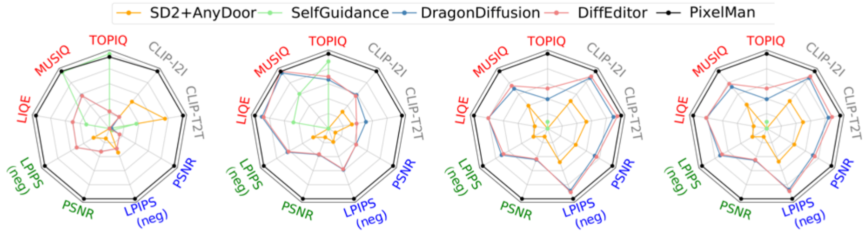
Figure: Visual comparison examples (on COCOEE dataset).

Evaluation – Editing Quality and Efficiency

PixelMan improves editing quality

- Object is consistent to the source (attributes and identity)
- Background is preserved after editing (texture and color)
- Original object is completed removed and inpainted with cohesively background

IQA, Object Consistency, Background Consistency, Semantic Consistency



(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

- PixelMan (@ 16 steps) has better quality than competitive training-free and training-based methods (@ 50 steps)
- Consistently outperform other methods at 8,16,50 steps (when using the same #Steps)
- Superior quality in 4 evaluation aspects with 9 metrics

Method	Efficiency			Image Quality Assessment			Object Consistency		Background Consistency		Semantic Consistency	
	# Steps	# NFEs ↓	Latency (secs) ↓	TOPIQ ↑	MUSIQ ↑	LIQE ↑	LPIPS ↓	PSNR ↑	LPIPS ↓	PSNR ↑	CLIP-T2T ↑	CLIP-I2I ↑
SDv2-Inpainting+AnyDoor	50	100	15	0.549	67.61	3.98	0.068	24.28	0.172	21.52	0.905	0.934
Self-Guidance	50	100	11	0.554	65.91	3.90	0.083	22.77	0.259	17.86	0.865	0.897
DragonDiffusion	50	160	23	0.571	68.87	4.27	0.034	28.59	0.098	23.99	0.933	0.965
DiffEditor	50	176	24	0.579	69.09	4.27	0.036	28.49	0.094	24.23	0.937	0.967
PixelMan	16	64	9	**0.605**	**69.98**	**4.35**	**0.015**	**35.62**	**0.074**	**26.43**	**0.946**	**0.974**
SDv2-Inpainting+AnyDoor	50	100	15	0.549	67.61	3.98	0.068	24.28	0.172	21.52	0.905	0.934
Self-Guidance		100	11	0.554	65.91	3.90	0.083	22.77	0.259	17.86	0.865	0.897
DragonDiffusion		160	23	0.571	68.87	4.27	0.034	28.59	0.098	23.99	0.933	0.965
DiffEditor		176	24	0.579	69.09	4.27	0.036	28.49	0.094	24.23	0.937	0.967
PixelMan		206	27	**0.605**	**70.17**	**4.36**	**0.014**	**35.92**	**0.077**	**26.28**	**0.941**	**0.974**
SDv2-Inpainting+AnyDoor	16	32	5	0.556	67.66	3.93	0.067	24.44	0.172	21.60	0.914	0.933
Self-Guidance		32	4	0.600	69.07	4.13	0.083	22.85	0.195	21.02	0.899	0.916
DragonDiffusion		64	9	0.588	69.92	4.31	0.040	27.58	0.124	23.34	0.923	0.950
DiffEditor		58	9	0.590	**69.99**	4.30	0.041	27.52	0.125	23.34	0.917	0.949
PixelMan		64	9	**0.605**	69.98	**4.35**	**0.015**	**35.62**	**0.074**	**26.43**	**0.946**	**0.974**
SDv2-Inpainting+AnyDoor	8	16	3	0.556	66.86	3.78	0.068	24.50	0.177	21.49	0.916	0.929
Self-Guidance		16	2	**0.604**	69.58	3.95	0.085	22.72	0.232	21.73	0.900	0.892
DragonDiffusion		32	5	0.567	68.45	4.05	0.050	26.84	0.186	22.31	0.886	0.908
DiffEditor		32	5	0.567	68.44	4.05	0.050	26.86	0.186	22.31	0.885	0.908
PixelMan		28	4	0.602	**69.63**	**4.32**	**0.016**	**35.33**	**0.071**	**26.70**	**0.926**	**0.971**

Evaluation – Detailed Results

Method	Efficiency			Image Quality Assessment			Object Consistency		Background Consistency		Semantic Consistency	
	# Steps	# NFEs ↓	Latency (secs) ↓	TOPIQ ↑	MUSIQ ↑	LIQE ↑	LPIPS ↓	PSNR ↑	LPIPS ↓	PSNR ↑	CLIP-T2T ↑	CLIP-I2I ↑
SDv2-Inpainting+AnyDoor	50	100	16	0.621	71.19	4.22	0.052	26.06	0.159	21.21	0.866	0.907
Self-Guidance	50	100	14	0.586	69.41	3.61	0.064	24.21	0.273	17.92	0.817	0.869
DragonDiffusion	50	160	30	0.690	**74.95**	4.72	0.030	29.68	0.083	25.38	**0.902**	0.934
DiffEditor	50	176	32	0.691	74.94	**4.73**	0.032	29.59	0.083	25.44	0.899	0.933
PixelMan	16	64	11	**0.696**	74.66	4.70	**0.015**	**35.90**	**0.070**	**27.18**	0.898	**0.939**
SDv2-Inpainting+AnyDoor		100	16	0.621	71.19	4.22	0.052	26.06	0.159	21.21	0.866	0.907
Self-Guidance		100	14	0.586	69.41	3.61	0.064	24.21	0.273	17.92	0.817	0.869
DragonDiffusion	50	160	30	0.690	**74.95**	4.72	0.030	29.68	0.083	25.38	**0.902**	0.934
DiffEditor		176	32	**0.691**	74.94	4.73	0.032	29.59	0.083	25.44	0.899	0.933
PixelMan		206	34	0.688	74.72	**4.75**	**0.015**	**36.26**	**0.073**	**26.74**	0.896	**0.940**
SDv2-Inpainting+AnyDoor		32	6	0.625	71.29	4.17	0.051	26.21	0.159	21.25	0.856	0.907
Self-Guidance		32	6	0.663	73.41	4.16	0.064	24.00	0.194	20.95	0.847	0.886
DragonDiffusion	16	64	12	**0.697**	**75.21**	**4.72**	0.033	29.19	0.104	24.99	0.894	0.917
DiffEditor		58	11	**0.697**	75.20	**4.72**	0.033	29.15	0.105	25.00	0.889	0.917
PixelMan		64	11	0.696	74.66	4.70	**0.015**	**35.90**	**0.070**	**27.18**	**0.898**	**0.939**
SDv2-Inpainting+AnyDoor		16	3	0.627	70.92	4.04	0.051	26.31	0.162	21.21	0.849	0.902
Self-Guidance		16	3	0.678	73.07	4.01	0.065	23.97	0.255	20.76	0.851	0.845
DragonDiffusion	8	32	6	0.692	**74.62**	4.46	0.038	28.57	0.173	22.68	0.856	0.876
DiffEditor		32	6	0.692	**74.62**	4.46	0.038	28.57	0.173	22.68	0.852	0.876
PixelMan		28	5	**0.695**	74.59	**4.67**	**0.016**	**35.57**	**0.067**	**27.74**	**0.900**	**0.937**

Table 4: **Quantitative results on the ReS (Yang et al. 2022) dataset.** Comparing PixelMan with other methods including Self-Guidance (Epstein et al. 2023), DragonDiffusion (Mou et al. 2024b), DiffEditor (Mou et al. 2024a), and the training-based SDv2-Inpainting+AnyDoor (Rombach et al. 2022; AI 2022b; Chen et al. 2024b) baseline. The ↓ indicates lower is better, and the ↑ means the higher the better. The best performance result is marked in **bold** and the second best result is annotated with underlines. Our reported latency measures the average wall-clock time over ten runs for generating one image on this dataset in seconds with a V100 GPU.

Evaluation – Detailed Results

Figure: Visual comparison on ReS dataset at 16 and 50 steps (PixelMan vs. Others)

Evaluation – Detailed Results

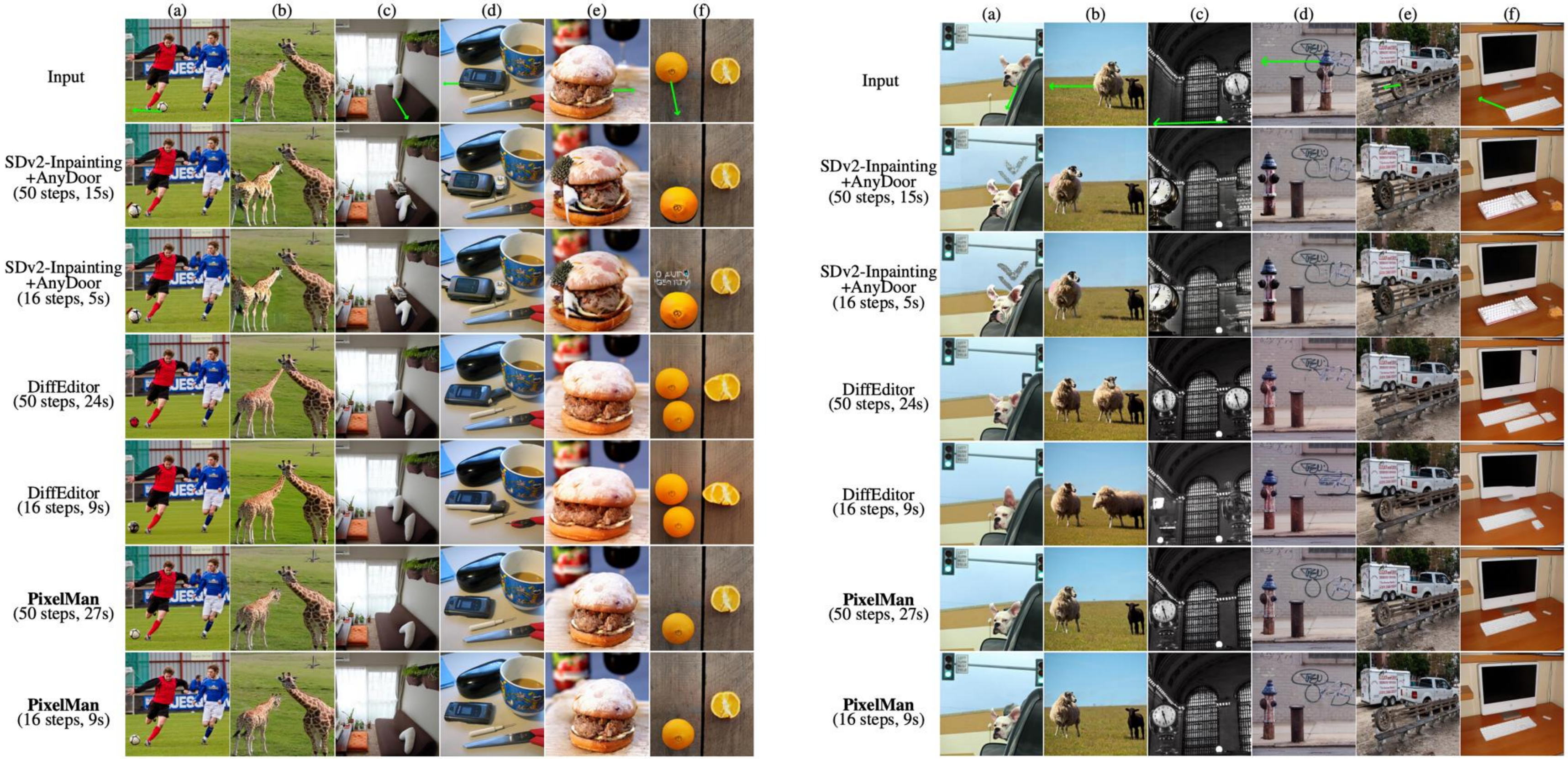


Figure: Visual comparison on COOCEE dataset at 16 and 50 steps (PixelMan vs. Others)

Evaluation – Detailed Results

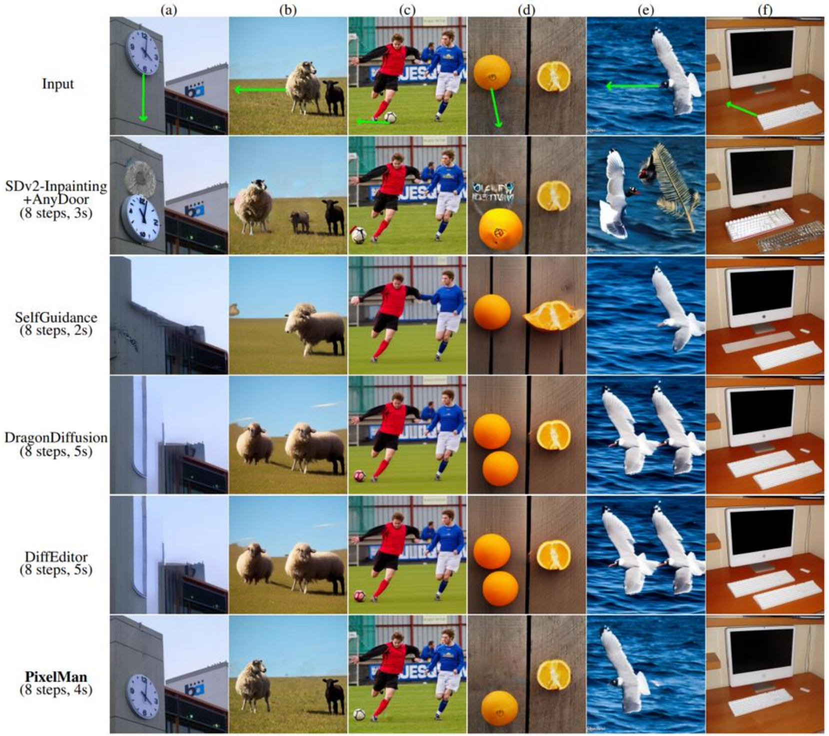


Figure: Visual comparison on COCOEE dataset at 8 steps (PixelMan vs. Others).

Figure: Visual comparison on ReS dataset at 8 steps (PixelMan vs. Others).

Evaluation – Other Consistent Object Editing Tasks

Consistent Object Editing Tasks

- Object Moving (Repositioning)
- **Object Pasting**
  - Source object is from a separate reference image
- Object Resizing
  - Object Enlarging
  - Object Shrinking

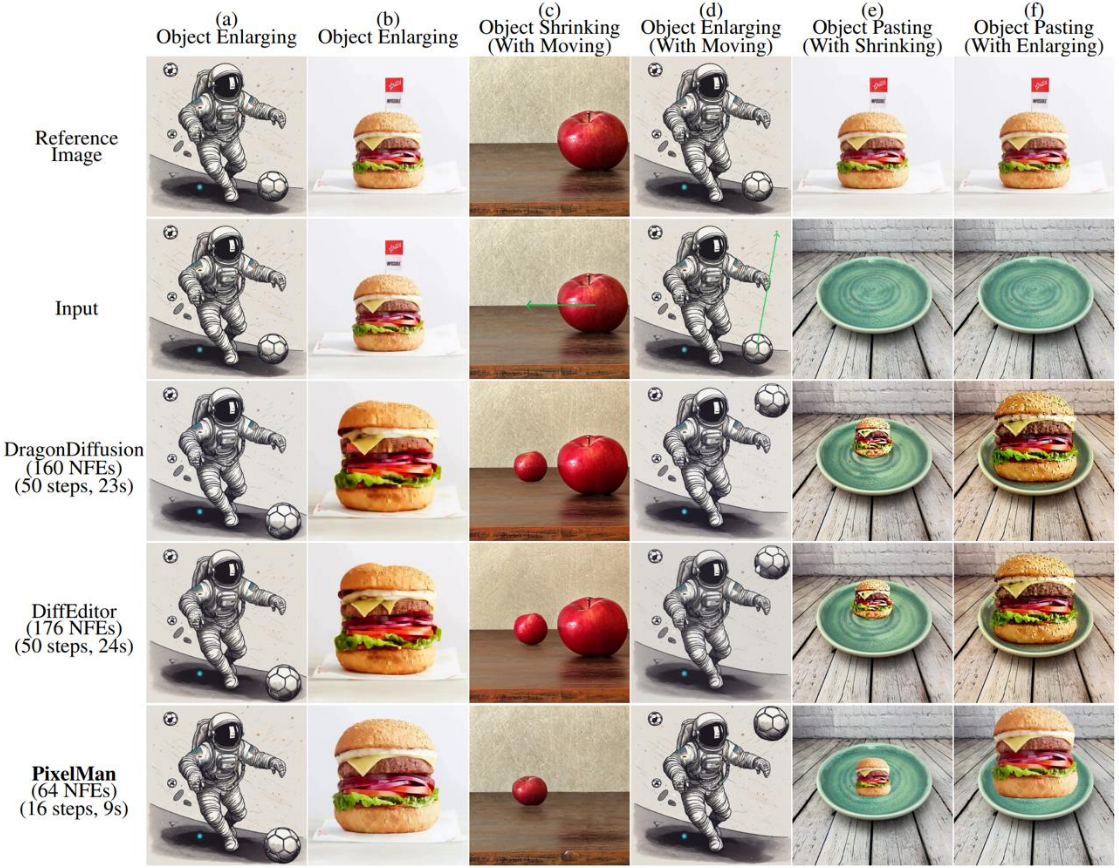
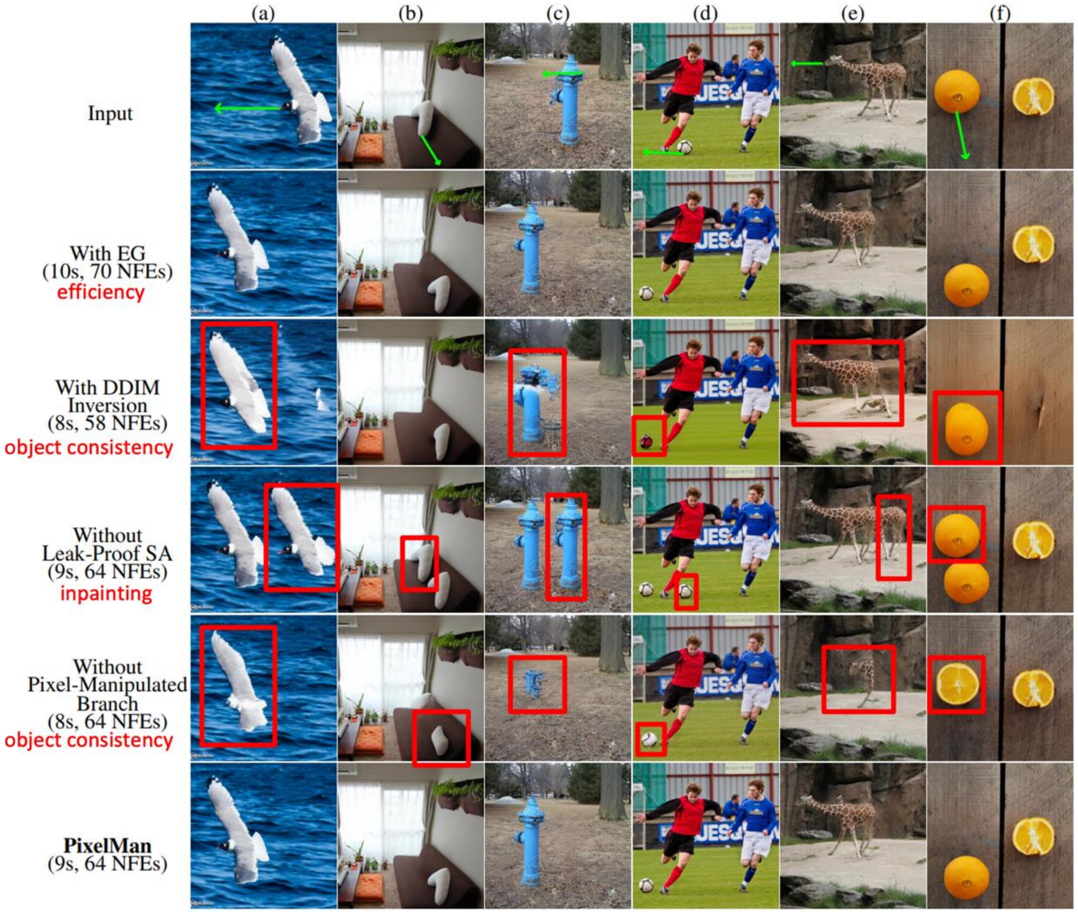


Figure 13: **Qualitative examples on other consistent object editing tasks** including object resizing, and object pasting.

Evaluation – Ablation Study

Ablation of the proposed method:

- Editing guidance with latent optimization (update z instead of ϵ), reduces #NFE while maintaining the quality
- The three-branched inversion-free sampling approach improves the object consistency over the DDIM inversion approach, while enabling high-quality editing in fewer steps
- The leak-proof SA mechanism significantly improve inpainting quality, completely inpaint the source object with cohesive background
- The pixel-manipulated anchor allows consistent reproduction of the object and background



Conclusion

- PixelMan is an inversion-free and training-free method for high quality consistent object editing. Our method improves editing quality and enables faster editing, outperforming methods requiring 50 steps with only 16 steps

- Our method preserves consistency in the object and background
  - We utilize pixel manipulation, i.e., duplicate the source object to the target location in pixel space to serve as consistency anchor
  - We design a three-branched sampling approach to compute the delta edit direction, enabling seamless harmonization with lighting, shadows, and edges
- By introducing a leak-proof self-attention technique, our method prevents attention leakage, ensuring cohesive inpainting of the original object location

- Validated on COCOEE and ReS datasets with superior performance in object, background, and semantic consistency metrics. Achieves higher or comparable overall image quality while reducing latency

AAAI-25 / IAAI-25 / EAAI-25
FEBRUARY 25 – MARCH 4, 2025 | PHILADELPHIA, USA

Thank you!

Liyao Jiang
PhD Student
University of Alberta
Email: liyao1@ualberta.ca

Links:

- Project Page: https://liyaojiang1998.github.io/projects/PixelMan/
- Paper: https://arxiv.org/abs/2412.14283

PixelMan@AAAI2025
liyaojiang1998.github.io