PixelMan@AAAI2025
liyaojiang1998.github.io

Keywords:
Generative AI, Diffusion Model, Image Editing, Non-Rigid Editing, Consistent Object Editing, High Consistency, High Efficiency, Inversion-Free, Training-Free

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

HUAWEI

UNIVERSITY OF ALBERTA

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*

Background

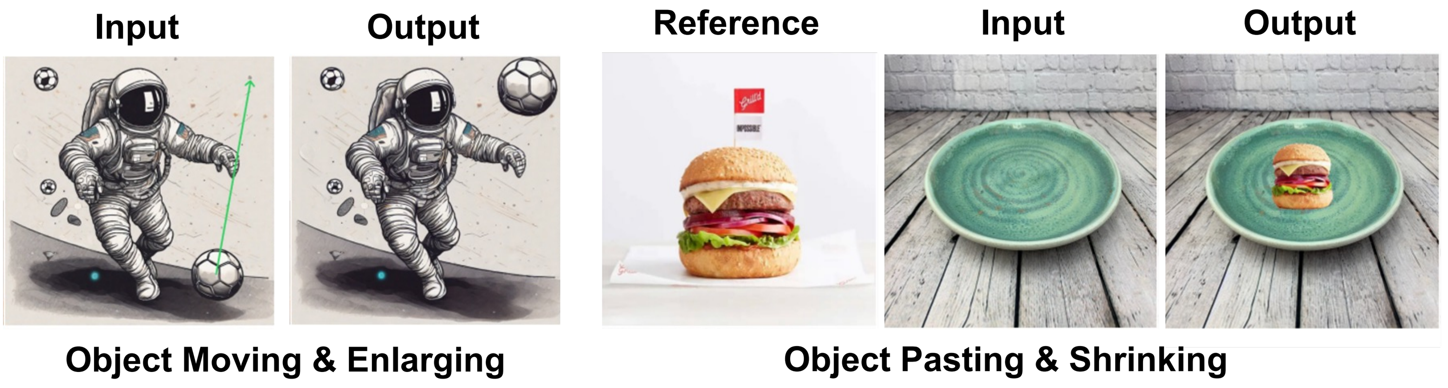
Promising results in text-guided rigid image editing (i.e., editing color, texture, attributes, and style)

Our focus: consistent object editing

- Preserve consistency for object and background
- Only edit non-rigid object attributes (e.g., position, size, composition)
- Applications: object repositioning (moving), resizing, pasting

A challenging task involving multiple sub-tasks

1. Faithful reproduction at the target location
2. Maintain background scene details
3. Seamless harmonization of new object and its surrounding
4. Inpainting the vacated area with cohesive background



Challenges

1. Low efficiency

- Rely on inversion, requiring many (e.g., at least 50) steps
- Compromising quality when steps are reduced

2. Low object and background consistency

- Altered object identity, inconsistent background

3. Incomplete & incoherent inpainting

- Fail to inpaint vacated area with cohesive background

Results

PixelMan improves editing quality

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

While having better efficiency

- PixelMan@16 steps outperforms other methods@50steps
  - Reduce latency: 24s -> 9s; Reduce #NFEs: 176 -> 64
- Consistently outperform other methods when using the same #Steps (at 8,16,50 steps)

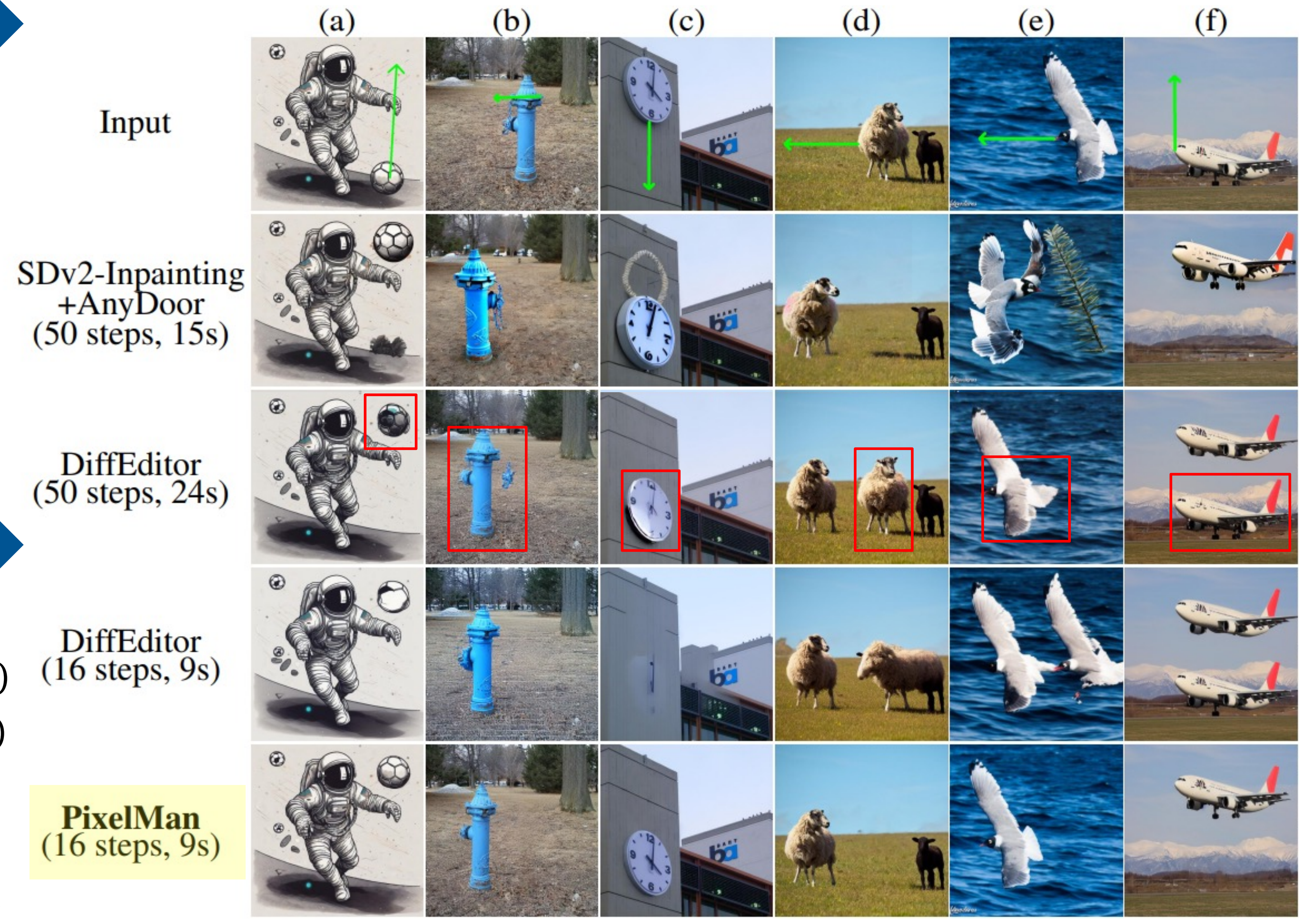
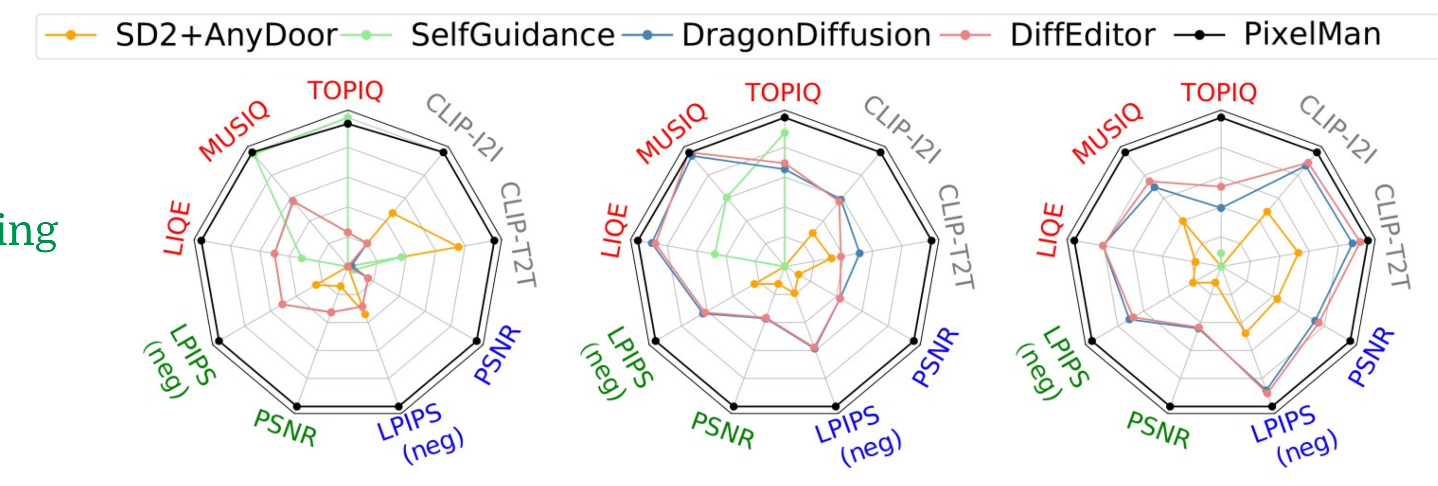


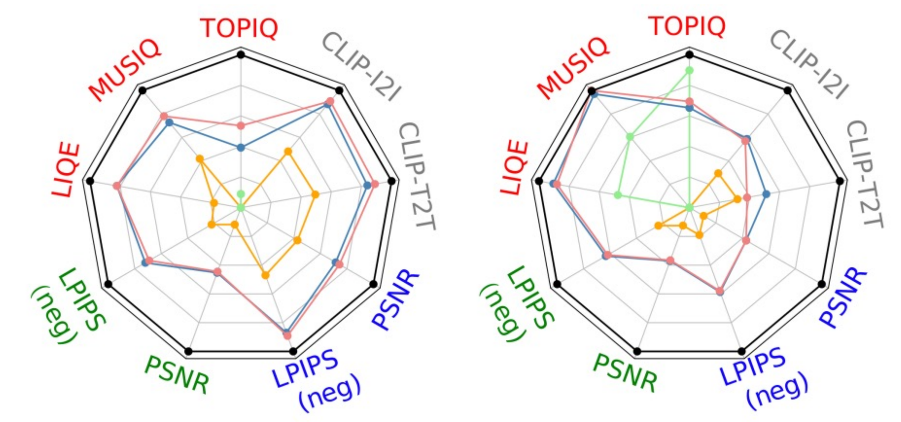
Figure: Visual comparison examples (on COCOEE dataset).



(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

Quantitative Evaluation Aspects:
IQA, Object Consistency, Background Consistency, Semantic Consistency

Methodology

1. Three-branched inversion-free sampling

- Pixel manipulation: preserve consistency
- Inversion-free: improve efficiency

Three-branches:

a) Pixel-manipulated branch: copy to target location in pixel space
b) Target branch: anchor target latents to pixel-manipulated latents at each step
c) Source branch: preserve clean K, V as context to enhance harmonization (e.g., lighting, shadow, edge blending)

2. Editing guidance techniques

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

Output = Anchor + Delta Edit Direction x Mask

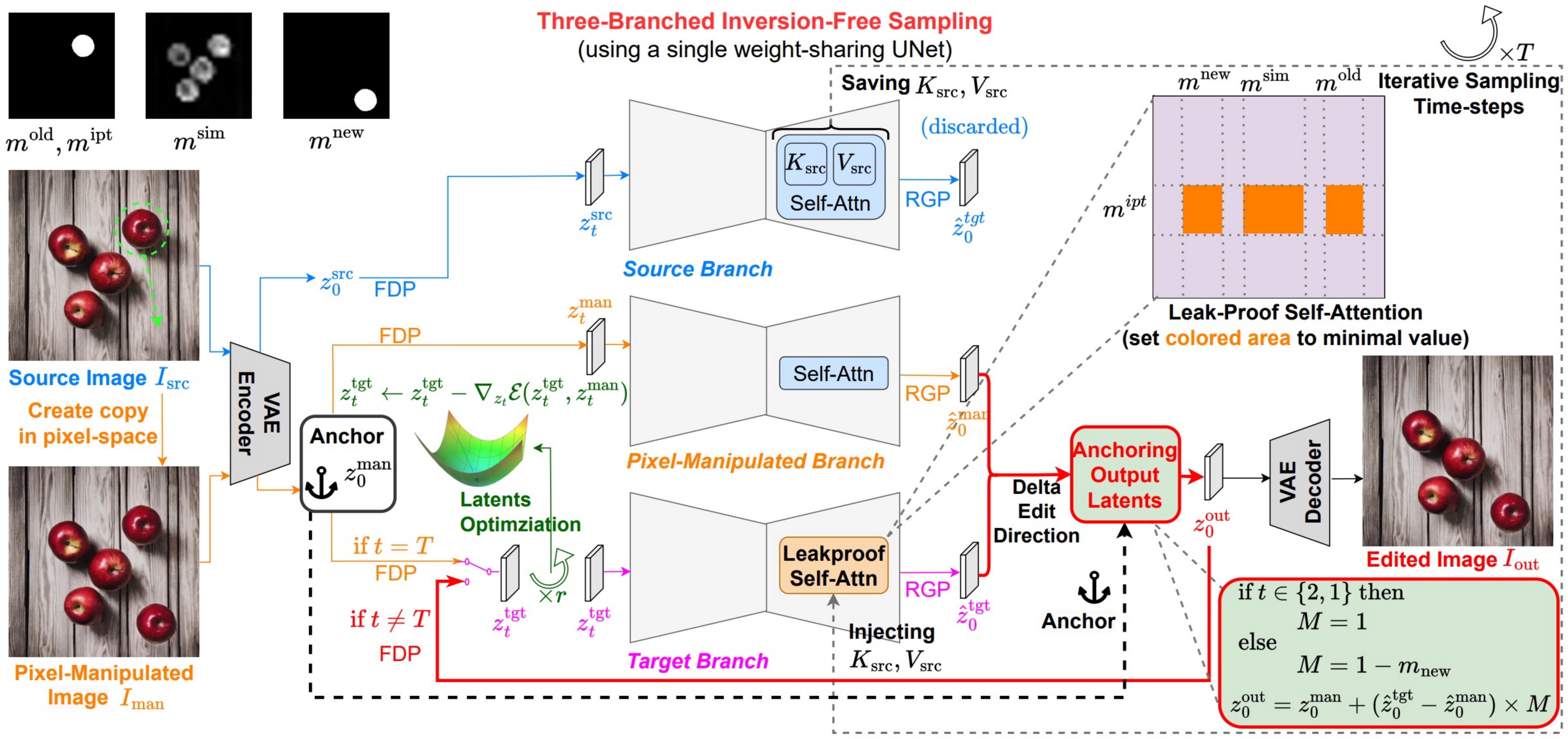
Generation: find delta edits for harmonization and inpainting on top of the anchor

a) Use energy functions with latents optimization (updates z instead of ϵ), reducing NFE
b) Injection of source K, V into the target branch
c) Apply leak-proof self-attention in target branch

3. Leak-proof self-attention

To achieve complete and cohesive inpainting

- Root cause of inpainting failure
  - Information leakage from similar objects through self-attention
- Leak-proof self-attention: prevent attention to source, target, and similar objects
  - Set the corresponding QKˆT elements to minimal values



Conclusion

- PixelMan is an inversion-free and training-free method
  - Achieve high-quality consistent object editing
  - Improve editing quality and enables faster editing
  - Surpass methods requiring 50 steps with only 16 steps
- Superior performance in object, background, and semantic consistency metrics on COCOEE and ReS datasets
- Achieve higher or comparable overall image quality while reducing latency