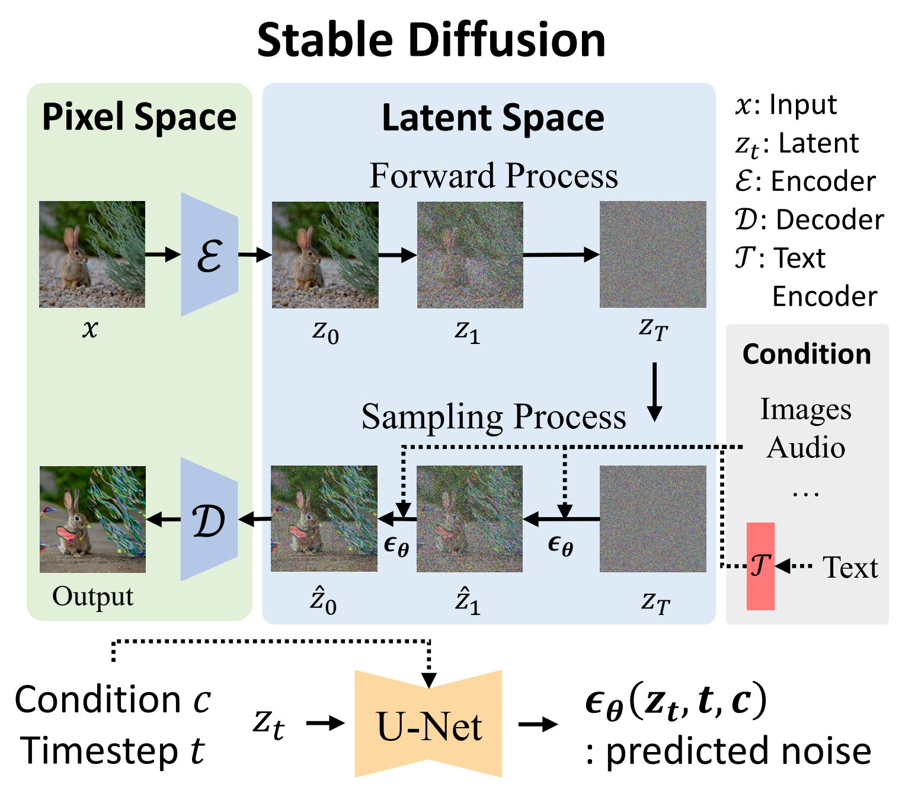
AdvPaint: Protecting Images from Inpainting Manipulation via Adversarial Attention Disruption

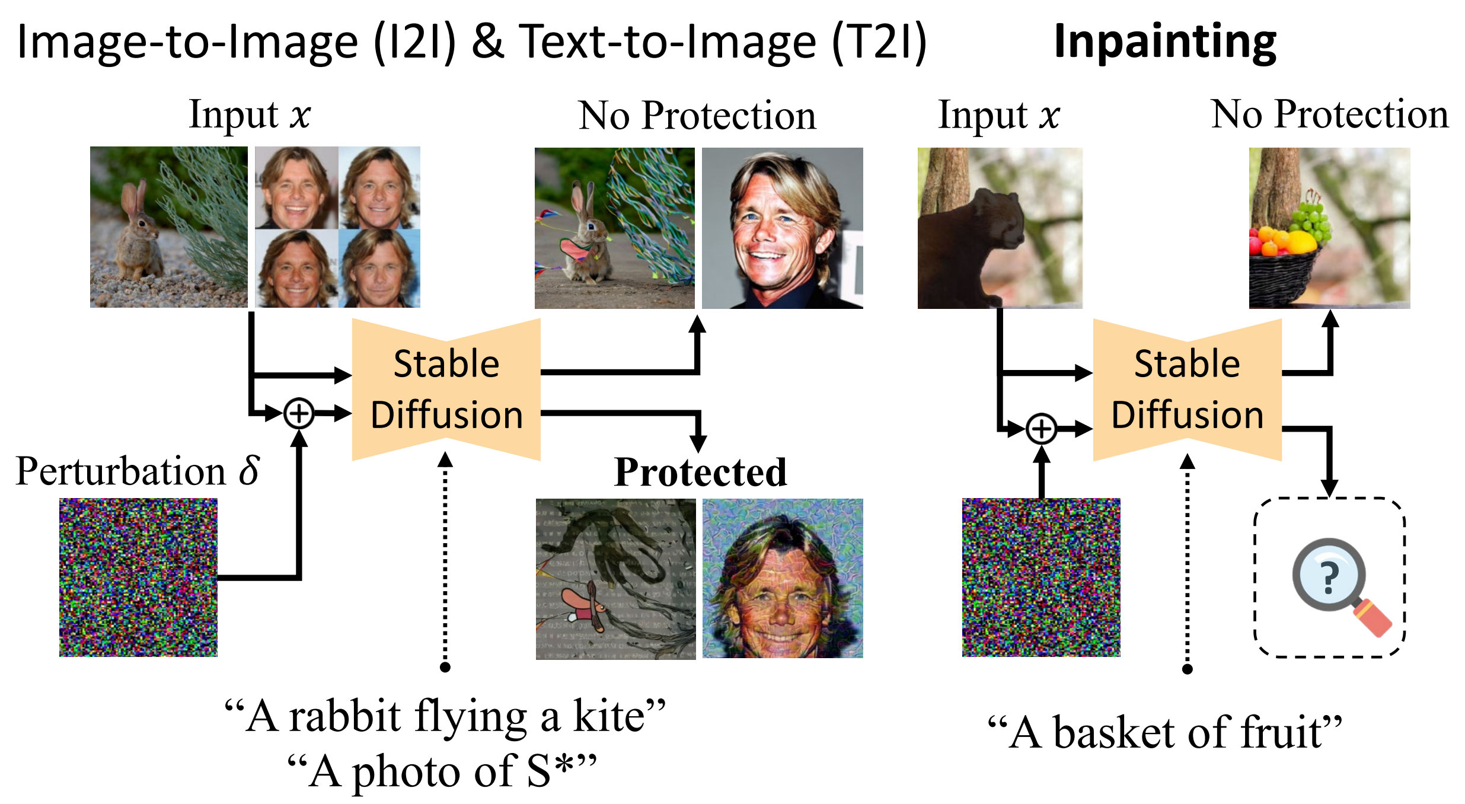
Joonsung Jeon, Woo Jae Kim, Suhyeon Ha, Sooel Son*, and Sung-Eui Yoon*

Project Page

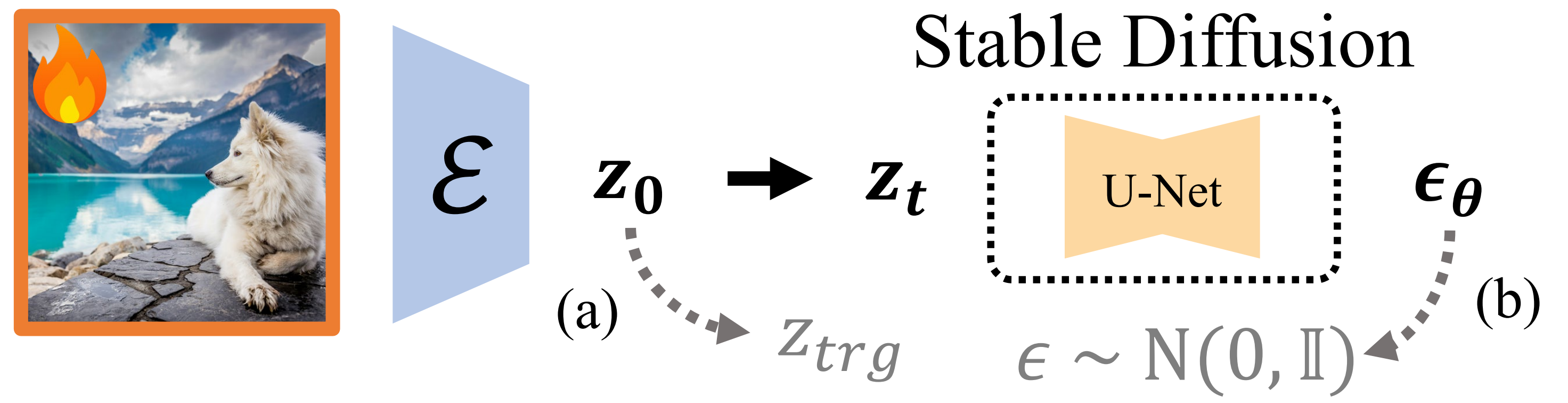
This was work supported by NRF (No. RS-2023-00208506) and IITP (No. RS-2020-II200153).

I. Motivation

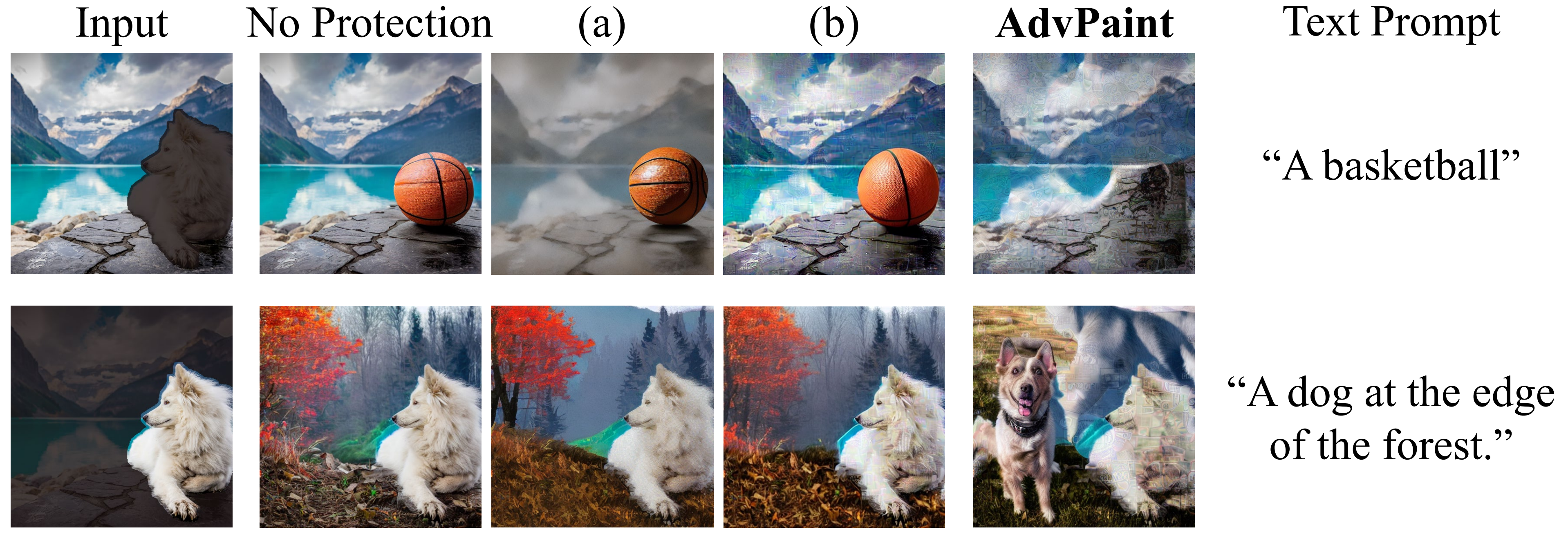




- ✓ **Adversarial perturbation δ** is utilized to protect images from unauthorized manipulations (i.e., *image-to-image, text-to-image*).
- ✓ However, preventing unauthorized *inpainting* has been *rarely assessed*.

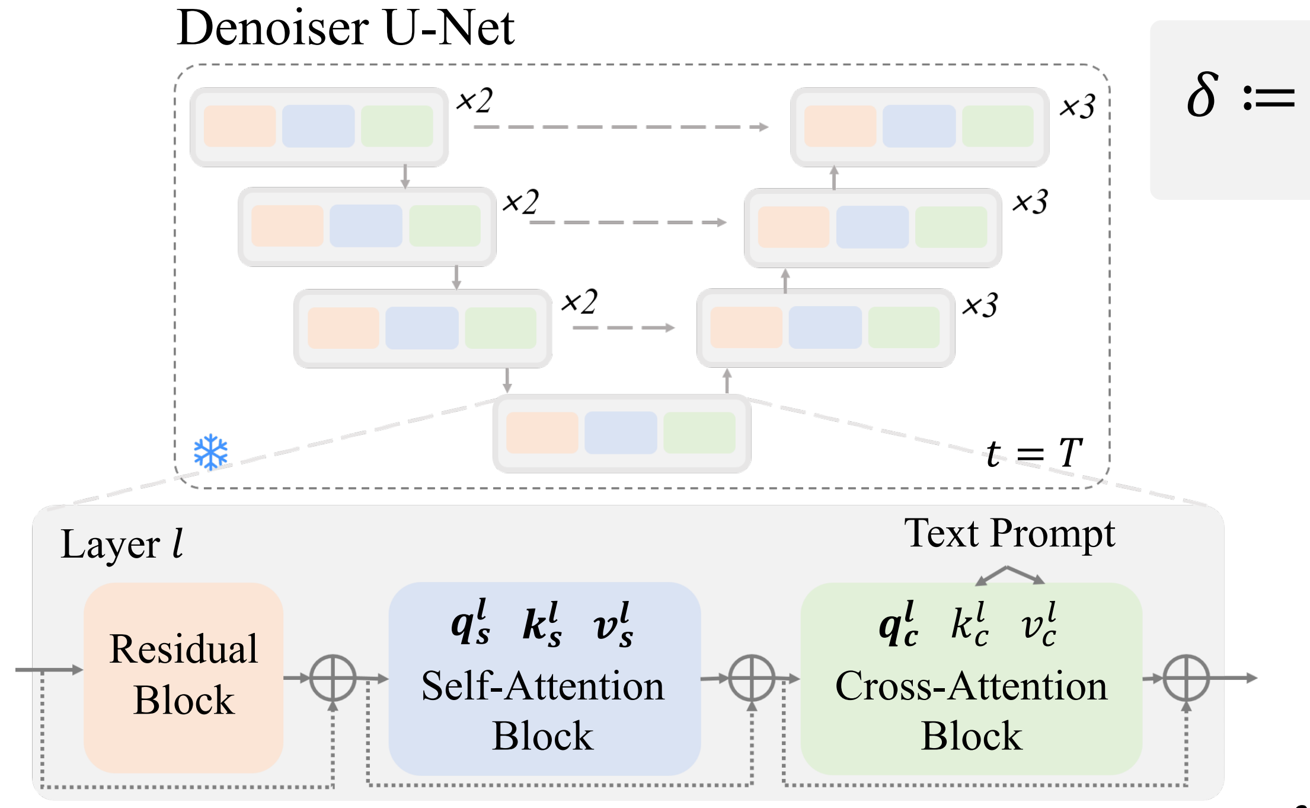


- ✓ **Problem #1**: Unlike *I2I* or *T2I*, some regions of perturbation are covered by the given mask in *inpainting* tasks.
- ✓ **Problem #2**: Baselines - (a), (b)
  - Only utilize a single perturbation
  - Only shift the latent representation
    - Overlooking the ***implicit steps*** within the generation process



II. Methodology

1 Adversarial Attack on Attention Mechanism



$$\delta := arg \max_{\|\delta\|_\infty \le \eta} \boldsymbol{\mathcal{L}_{attn}} = arg \max_{\|\delta\|_\infty \le \eta} (\mathcal{L}_{cross} + \mathcal{L}_{self})$$

$$\mathcal{L}_{cross} = \sum_l \left\| Q_c^l(\phi(x+\delta)) - Q_c^l(\phi(x)) \right\|^2$$

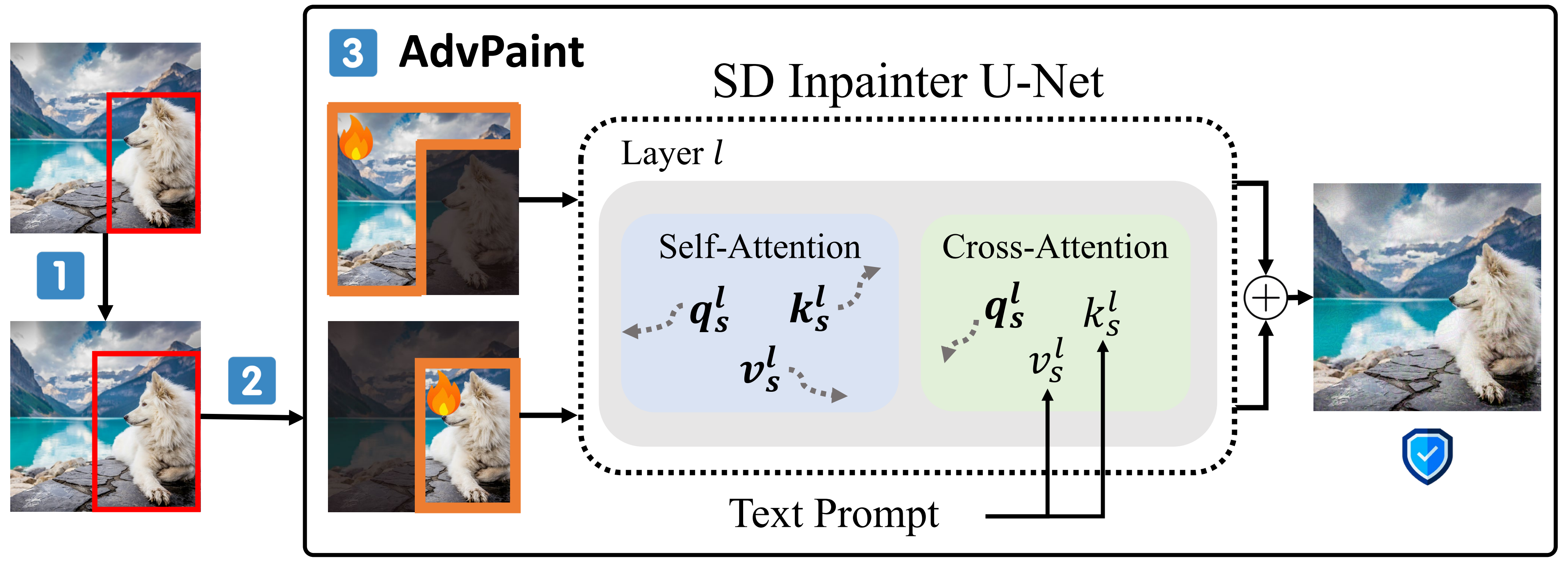
$$\mathcal{L}_{self} = \sum_l \begin{pmatrix} \left\| Q_s^l(\phi(x+\delta)) - Q_s^l(\phi(x)) \right\|^2 \\ + \left\| K_s^l(\phi(x+\delta)) - K_s^l(\phi(x)) \right\|^2 \\ + \left\| V_s^l(\phi(x+\delta)) - V_s^l(\phi(x)) \right\|^2 \end{pmatrix}$$

$\boldsymbol{x}$: input image $\boldsymbol{\delta}$: perturbation ϕ: feature extractor
Q, K, V: linear projection operator of $\boldsymbol{q}, \boldsymbol{k}, \boldsymbol{v}$

- ✓ Attention Mechanism
  - **Self-attention blocks**: understand the semantics & spatial structure
  - **Cross-attention blocks**: align the generation with the external condition

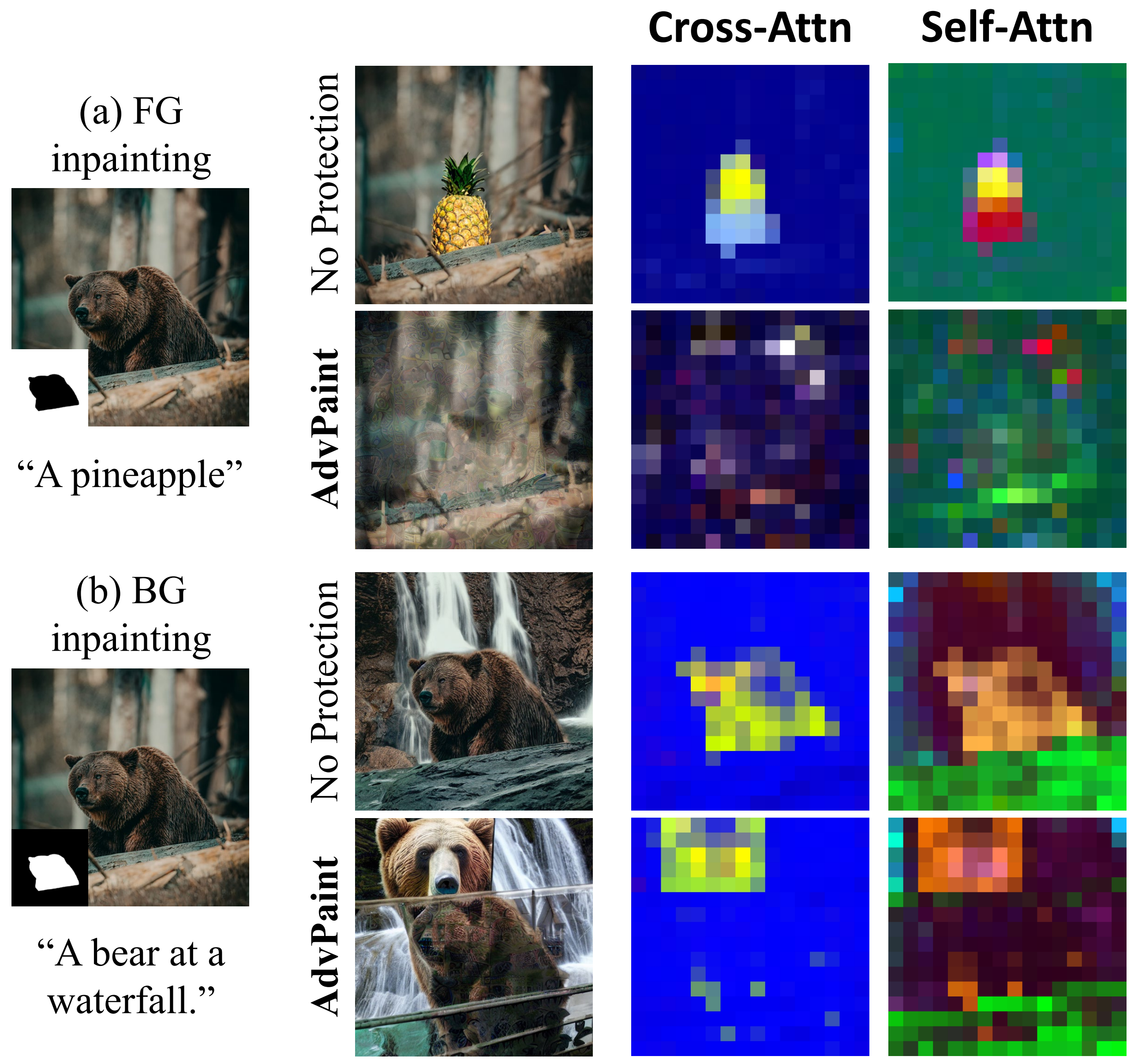
2 Two-stage Optimization

- ✓ Apply separate protection for *objects* and *BG*
  1. Enlarge the bounding-box to fully cover the object
  2. Separate fore- (FG) and back-ground (BG) with the bounding-box
  3. Optimize **twice** via $\boldsymbol{\mathcal{L}_{attn}}$



III. Experiments

1) Attention Maps



2) Results

	Optimization Methods	Foreground Inpainting						Background Inpainting					
		m^{seg}			m^{bb}			m^{seg}			m^{bb}		
		FID ↑	Prec ↓	LPIPS ↑	FID ↑	Prec ↓	LPIPS ↑	FID ↑	Prec ↓	LPIPS ↑	FID ↑	Prec ↓	LPIPS ↑
A	Photoguard	230.49	0.5244	0.6494	185.86	0.7212	0.6236	118.85	0.4332	0.4141	132.51	0.1844	0.5220
	AdvDM	232.39	0.3030	0.5287	181.13	0.4794	0.5231	94.49	0.5772	0.3111	116.60	0.2420	0.4191
	Mist	235.81	0.4590	0.5541	191.00	0.6490	0.5421	123.48	0.4004	0.3852	155.57	0.1602	0.5016
	CAAT	232.83	0.3430	0.5274	181.21	0.5314	0.5192	98.22	0.5414	0.3199	118.68	0.2382	0.4182
	SDST	212.90	0.5658	0.5042	174.85	0.7244	0.4994	112.17	0.4406	0.3841	133.15	0.2054	0.4809
B	SD Inpainter + $\min_\delta$ Eq.(a)	211.35	0.5644	0.5780	180.40	0.7214	0.5894	128.01	0.4006	0.4745	146.39	0.1374	0.5914
	SD Inpainter + $\max_\delta$ Eq.(b)	224.81	0.3860	0.4705	199.37	0.5186	0.4878	116.60	0.4832	0.3844	142.37	0.2078	0.4795
	SD Inpainter + $\min_\delta$ Eq.(b)	182.12	0.6124	0.5267	154.27	0.7560	0.5273	97.44	0.5852	0.386	107.43	0.2692	0.4902
	ADVPAINT	**347.88**	**0.0570**	**0.6731**	**289.63**	**0.1536**	**0.6762**	**219.07**	**0.2148**	**0.5064**	**303.90**	**0.0936**	**0.6105**

C Stage	Foreground Inpainting				Background Inpainting			
	m^{seg}		m^{bb}		m^{seg}		m^{bb}	
	FID ↑	Prec ↓	FID ↑	Prec ↓	FID ↑	Prec ↓	FID ↑	Prec ↓
1	345.76	0.0628	271.73	0.2056	191.15	0.2418	266.00	0.0938
2	**347.88**	**0.0570**	**289.63**	**0.1536**	**219.07**	**0.2148**	**303.90**	**0.0936**

- ✓ Comparison with
  - A **Previous methods**
  - B **Previous objectives (⇄ 1)**
  - C **Single perturbation (⇄ 2)**