

Taming Diffusion Prior for Image Super-Resolution with Domain Shift SDEs

Qinpeng Cui¹², Yixuan Liu¹, Xinyi Zhang², Qiqi Bao², Qingmin Liao², Li Wang¹, Tian Lu¹, Zicheng Liu¹, Zhongdao Wang^{†2}, Emad Barsoum¹

¹Advanced Micro Devices Inc.

²Tsinghua University

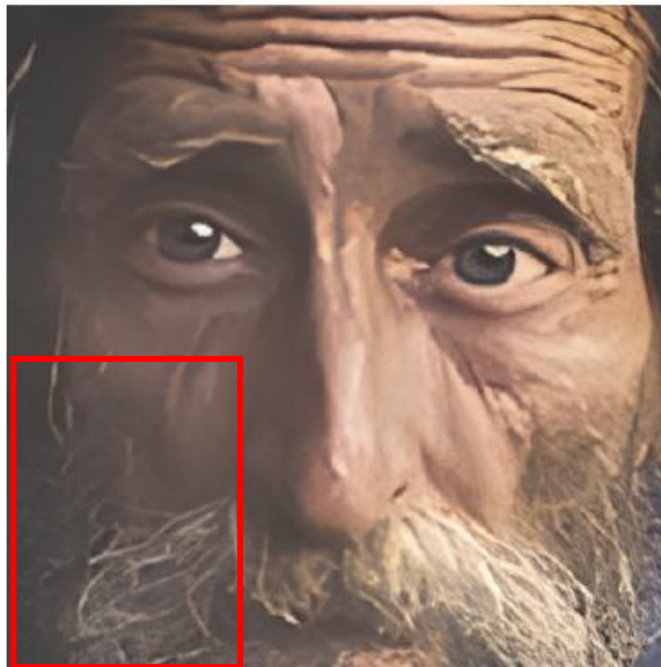


Motivation

- ❑ Image Super Resolution(SR): Traditional models aim to learn a mapping from low-resolution (LR) images to high-resolution(HR) image.
- ❑ Diffusion-based SR models have attracted substantial interest due to their powerful image restoration capabilities. However , prevailing diffusion models often struggle to strike an optimal balance between ***efficiency and performance***.



LR image



Previous SR method



SD-based model

Motivation

❑ Currently, diffusion-based SR strategies can be broadly categorized into two approaches:

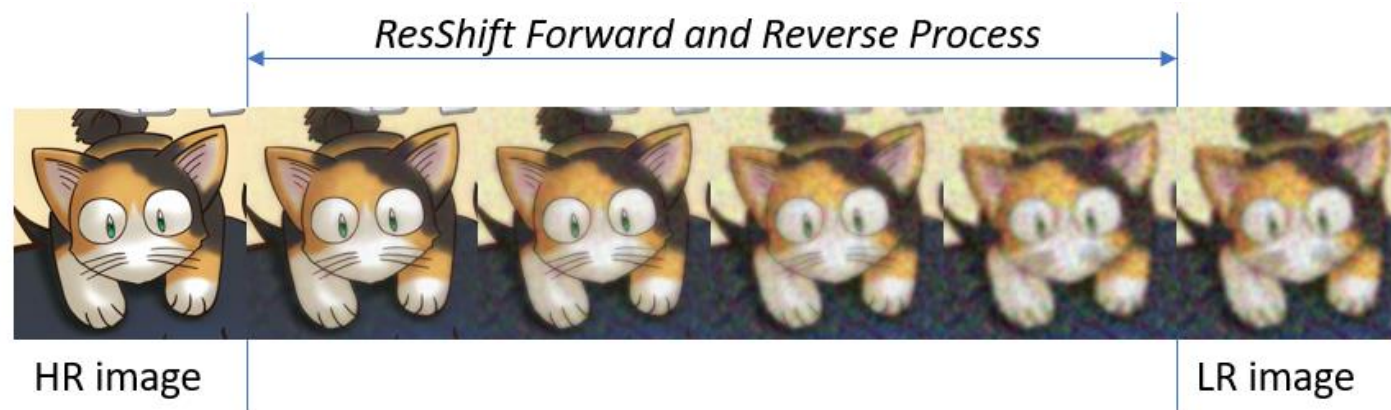
- 1) Follow the traditional diffusion process (High resolution images $\leftrightarrow$ Random Gaussian Noise):
 - Achieves good performance: leverage large-scale pretrained models(e.g. Stable Diffusion) as generative prior
 - Efficiency limitation due to the long transition path (start from noise)
 - eg: StableSR(IJCV'24)



Motivation

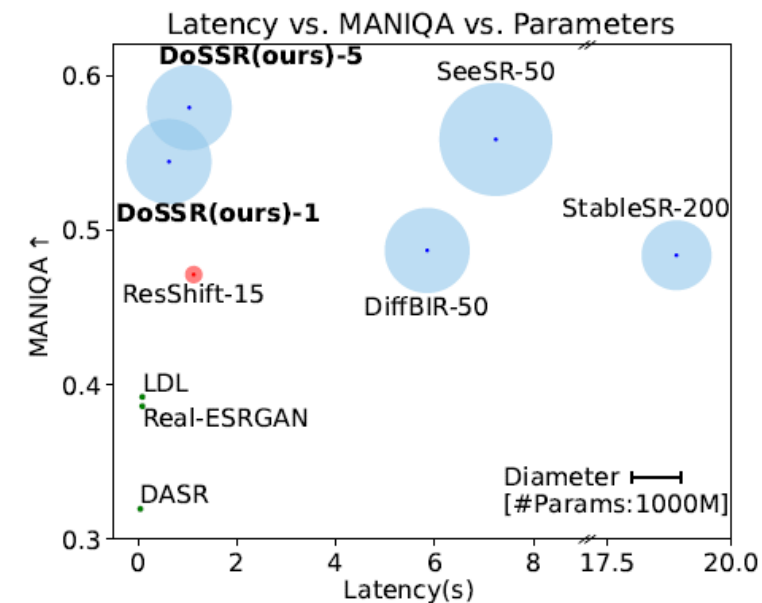
❑ Currently, diffusion-based SR strategies can be broadly categorized into two approaches:

- 1) Follow the traditional diffusion process (High resolution images $\leftrightarrow$ Random Gaussian Noise):
 - Achieves good performance: leverage large-scale pretrained models(e.g. Stable Diffusion) as generative prior
 - Efficiency limitation due to the long transition path (start from noise)
 - eg: StableSR(IJCV'24)
- 2) Redefine the diffusion process (High resolution images $\leftrightarrow$ Low resolution image):
 - poorer results: retraining a model from scratch for the SR task
 - Faster: start from LR image rather than gauss noise
 - eg: ResShift (NeurIPS' 23), FlowIE(CVPR' 24)

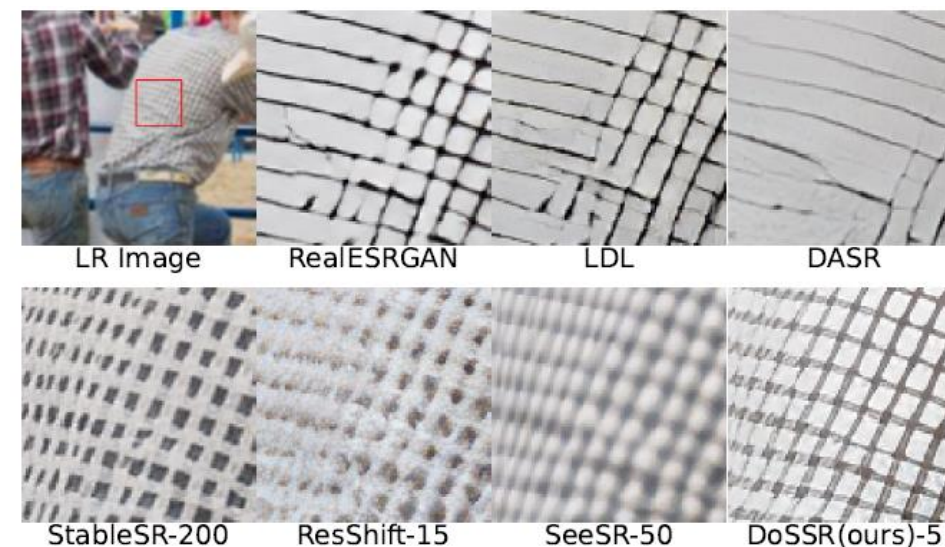


Contributions

- ❑ **Domain shift SDEs (DoS-SDE):** We propose a novel diffusion equation that improves efficiency while keeping good generative capability
 - Leveraging Stable Diffusion prior
 - Denoise starting from LR images rather than random Gaussians
- ❑ **Solvers for DoS-SDEs:** We design customized fast sampler, resulting in even higher efficiency



	Leverage SD Prior	Start from LR	Generative Capability	Efficiency
Category1 (StableSR)	✓		Good	Slow
Category2 (ResShift)		✓	Bad	Fast
DoSSR (ours)	✓	✓	Good	Fast



Method(*Domain Shift Equation*)

- ❑ Consider the SR task as a *gradual shift* from the *source domain* to the *target domain*.
 - source domain: the distribution of LR images $p_{data}(\hat{x}_0)$
 - target domain: the distribution of HR images $p_{data}(x_0)$
- ❑ How to describe this transition? Just *linear interpolation* is a simple and effective method !

$$\mathcal{D}(\hat{x}_0, x_0) = \eta_t \hat{x}_0 + (1 - \eta_t) x_0, \quad 0 \leq \eta_t \leq 1, \quad t = 1, 2, \dots, T,$$

where drift coefficient η_t monotonically non-decreases with timestep t .

- ❑ To enable linear combination, we can interpolate $\hat{x}_0$ to match the same dimensions as x_0 if necessary.
 - this operation applies similarly to the *latent space* as well

Method(*Diffusion Process with Domain Shift*)

- ❑ Combine this domain shift with the *Sable Diffusion* forward diffusion equation ?

- *replace the “noise-added object” in diffusion equation*
- *add noise while shifting from the HR to LR image distribution*

$$q(x_t|x_0) = \mathcal{N}(x_t; \alpha_t x_0, \sigma_t^2 \mathbf{I}), t = 1, 2, \dots, T, \longrightarrow q(x_t|x_0, \hat{x}_0) = \mathcal{N}(x_t; \alpha_t \mathcal{D}(\hat{x}_0, x_0) \sigma_t^2 \mathbf{I}), t = 1, 2, \dots, T,$$

where $\alpha_t, \sigma_t \geq 0$ and $\alpha_t^2 + \sigma_t^2 = 1$, which are called *noise schedule*, $\mathbf{I}$ is identity matrix

- ❑ Maximally preserve the *diffusion prior* by keeping the *noise schedule* unchanged.
 - rearranging the noise schedule requires significant training cost and can disrupt the pretrained model
- ❑ However, keeping the *noise schedule* unchanged ensures *the noising process endpoint* remains approximately *Gaussian noise*, as in Stable Diffusion, but we **aim to infer from LR**(+ little noise)
 - when $t = T, \alpha_T \rightarrow 0, \sigma_T \rightarrow 1, x_T \rightarrow N(0; \mathbf{I})$

Solve this by designing the shifting sequence $\{\eta_t\}_{t=1}^T$!

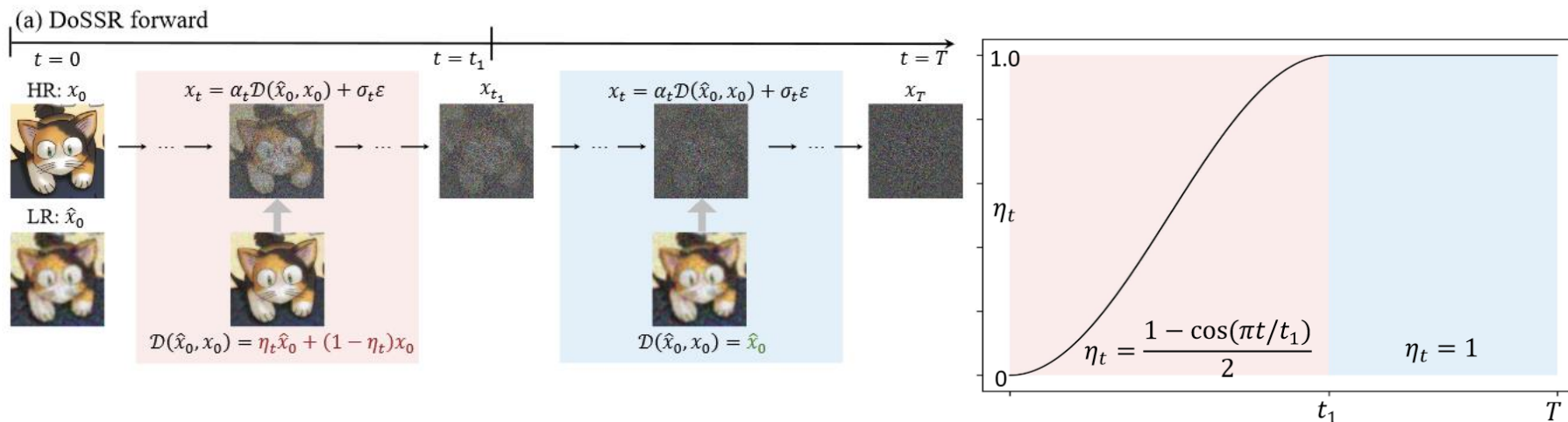
Method(*Shifting Sequence Design*)

□ Given that LR and noise are **known** prior distributions, we can use the shifting sequence to *shorten the unknown diffusion path length*.

□ Segmented shifting sequence in two parts:
$$\eta_t = \frac{1 - \cos(\pi \frac{t}{t_1})}{2} \text{ if } t \in [0, t_1], \quad \eta_t = 1 \text{ if } t \in [t_1, T].$$

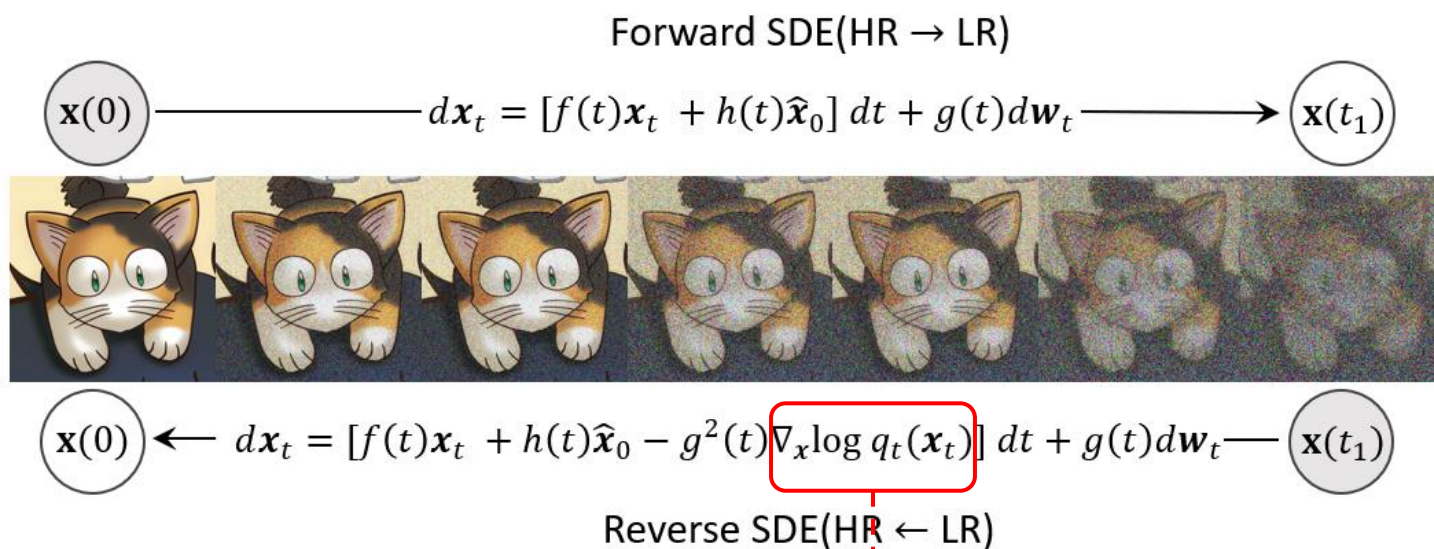
✓ Values of x_t for $t \in [t_1, T]$ are known and can be obtained through the forward process equation.

✓ Inference can start at time step t_1 instead of T



Method(*Diffusion DoS-SDEs*)

- Extend the discrete diffusion process to an SDE to enable efficient sampler design.



Domain Shift SDEs (DoS-SDEs)

- Transition is a linear Gaussian:

$$q(\mathbf{x}_t | \mathbf{x}_0, \hat{\mathbf{x}}_0) = \mathcal{N}(\mathbf{x}_t; \alpha_t(\eta_t \hat{\mathbf{x}}_0 + (1 - \eta_t)\mathbf{x}_0), \sigma_t^2 \mathbf{I}),$$

- Training by score matching:

$$\begin{aligned} \theta^* &= \arg \min_{\theta} \mathbf{E}_t \left\{ w(t) \mathbf{E}_{q_t(\mathbf{x}_t)} [|\epsilon_{\theta}(\mathbf{x}_t, \hat{\mathbf{x}}_0, t) + \sigma_t \nabla_{\mathbf{x}} \log q_t(\mathbf{x}_t)|] \right\} \\ &= \arg \min_{\theta} \mathbf{E}_t \left\{ w(t) \mathbf{E}_{q_0(\mathbf{x}_0)} \mathbf{E}_{q(\epsilon)} [|\epsilon_{\theta}(\mathbf{x}_t, \hat{\mathbf{x}}_0, t) - \epsilon|] \right\}, \end{aligned}$$

Method(Solvers for Diffusion DoS-SDEs)

❖ Design efficient samplers by solving the diffusion DoS-SDEs.

□ Exact solution of Diffusion DoS-SDEs:

$$\mathbf{x}_t = \frac{\alpha_t(1-\eta_t)}{\alpha_s(1-\eta_s)} \frac{\lambda_t^2}{\lambda_s^2} \mathbf{x}_s + \alpha_t(1-\eta_t) \left(\frac{\eta_t}{1-\eta_t} - \frac{\eta_s}{1-\eta_s} \frac{\lambda_t^2}{\lambda_s^2} \right) \hat{\mathbf{x}}_0$$

nonlinear integral term $-\alpha_t(1-\eta_t) \int_{\lambda_s}^{\lambda_t} \frac{2\lambda_t^2}{\lambda^3} \mathbf{x}_\theta(\mathbf{x}_\lambda, \hat{\mathbf{x}}_0, \lambda) d\lambda + \alpha_t(1-\eta_t) \sqrt{\lambda_t^2 - \frac{\lambda_t^4}{\lambda_s^2}} \mathbf{z}_s,$ where $\lambda_t = \frac{\sigma_t}{\alpha_t(1-\eta_t)}$ and $\mathbf{z}_s \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

first-order approximate(Euler method)

□ Sampler(approximate exact solution):

$$\tilde{\mathbf{x}}_t = \frac{\alpha_t(1-\eta_t)}{\alpha_s(1-\eta_s)} \frac{\lambda_t^2}{\lambda_s^2} \mathbf{x}_s + \underbrace{\alpha_t(1-\eta_t) \left(\frac{\eta_t}{1-\eta_t} - \frac{\eta_s}{1-\eta_s} \frac{\lambda_t^2}{\lambda_s^2} \right) \hat{\mathbf{x}}_0}_{\text{Domain Shift Guidance(DoSG)}} + \alpha_t(1-\eta_t) \left(1 - \frac{\lambda_t^2}{\lambda_s^2} \right) \mathbf{x}_\theta(\mathbf{x}_s, \hat{\mathbf{x}}_0, s) + \alpha_t(1-\eta_t) \sqrt{\lambda_t^2 - \frac{\lambda_t^4}{\lambda_s^2}} \mathbf{z}_s.$$

Domain Shift Guidance(DoSG)

Method(Solvers for Diffusion DoS-SDEs)

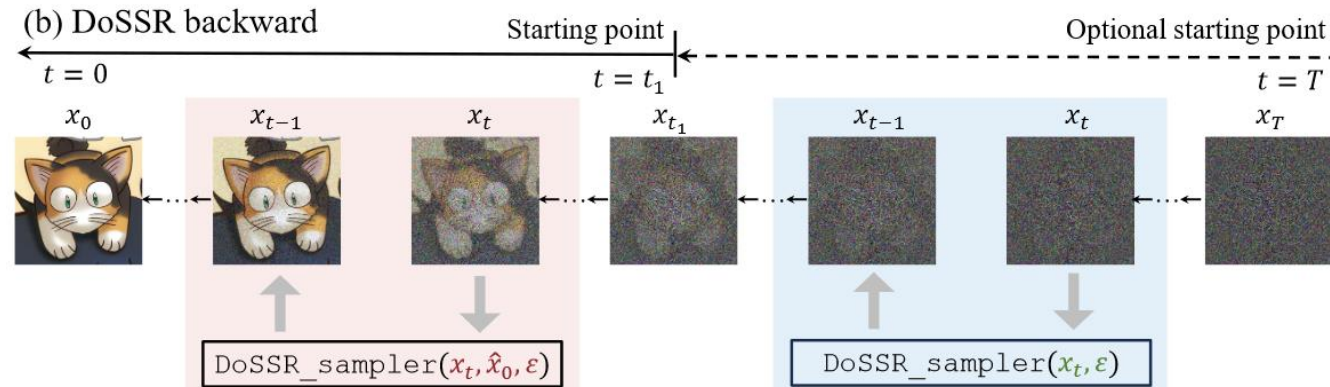
❖ Design efficient samplers by solving the diffusion DoS-SDEs.

□ Sampler(approximate exact solution):

$$\tilde{x}_t = \frac{\alpha_t(1-\eta_t)}{\alpha_s(1-\eta_s)} \frac{\lambda_t^2}{\lambda_s^2} x_s + \underbrace{\alpha_t(1-\eta_t) \left(\frac{\eta_t}{1-\eta_t} - \frac{\eta_s}{1-\eta_s} \frac{\lambda_t^2}{\lambda_s^2} \right) \hat{x}_0}_{\text{Domain Shift Guidance(DoSG)}} + \alpha_t(1-\eta_t) \left(1 - \frac{\lambda_t^2}{\lambda_s^2} \right) x_\theta(x_s, \hat{x}_0, s) + \alpha_t(1-\eta_t) \sqrt{\lambda_t^2 - \frac{\lambda_t^4}{\lambda_s^2}} z_s.$$

Domain Shift Guidance(DoSG)

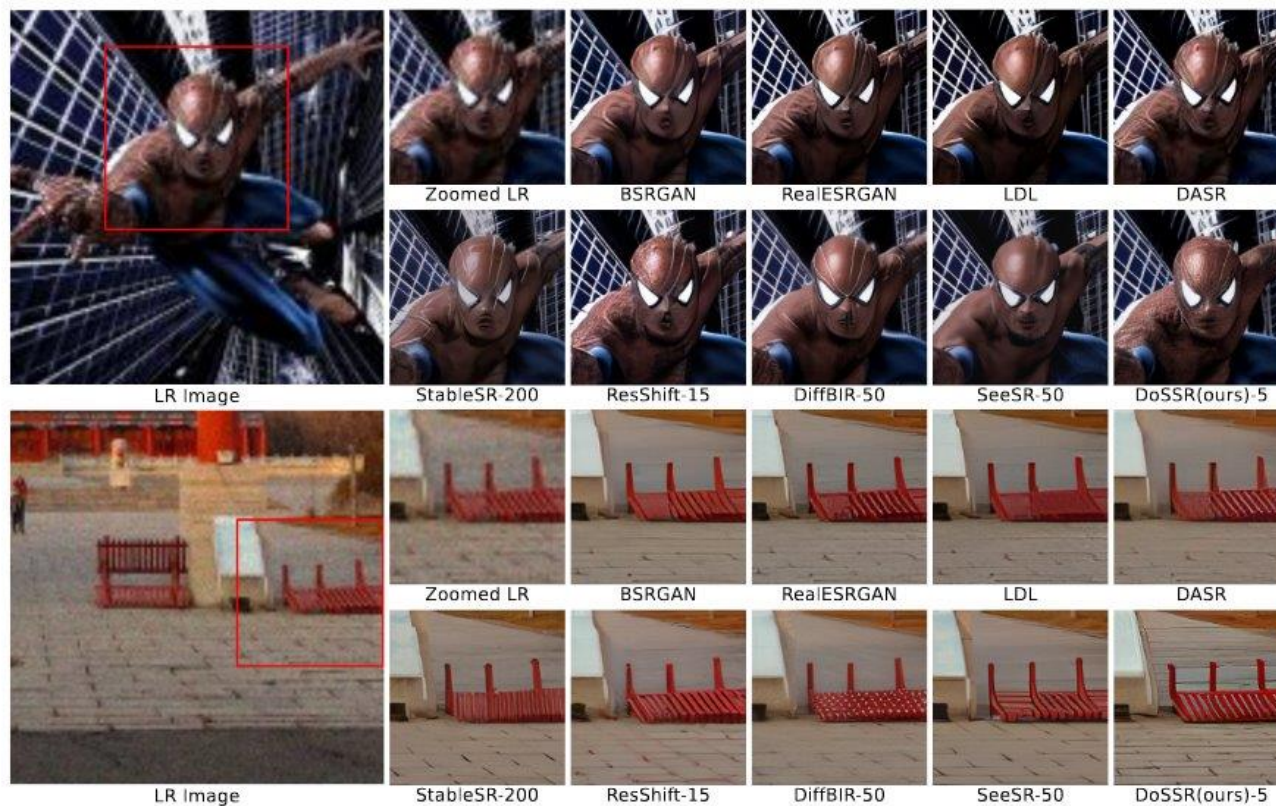
□ Inference:



Experiments

- ✓ SOTA performance on synthetic and real-world datasets.
- ✓ Requiring *only 5 sampling steps*, achieves a remarkable speedup of **5-7 times**.

Qualitative Comparison



Quantitative Comparison

Datasets	Metrics	BSRGAN	Real-ESRGAN	LDL	DASR	StableSR	ResShift	DiffBIR	SeeSR	DoSSR
DIV2k-Val	PSNR ↑	24.58	24.29	23.83	24.47	23.36	24.65	23.67	23.68	23.98
	SSIM ↑	0.6241	0.6338	0.6312	0.6277	0.5654	0.6148	0.5592	0.5987	0.6073
	LPIPS ↓	0.3351	0.3112	0.3256	0.3543	0.3114	0.3349	0.3516	0.3195	0.3371
	CLIPQA ↑	0.5246	0.5276	0.5179	0.5036	0.6771	0.6065	0.6693	0.6935	0.7014
	MUSIQ ↑	61.19	61.06	60.04	55.19	65.92	61.07	65.78	68.68	66.54
	MANIQA ↑	0.3547	0.3795	0.3736	0.3165	0.4193	0.4107	0.4568	0.5041	0.5294
RealSR	TOPIQ ↑	0.5456	0.5294	0.5142	0.4530	0.5974	0.5383	0.6142	0.6854	0.6766
	PSNR ↑	26.38	25.69	25.28	27.02	24.65	26.26	24.81	25.14	24.18
	SSIM ↑	0.7655	0.7615	0.7565	0.7714	0.7060	0.7404	0.6571	0.7194	0.6839
	LPIPS ↓	0.2656	0.2709	0.2750	0.3134	0.3002	0.3469	0.3607	0.3007	0.3374
	CLIPQA ↑	0.5114	0.4485	0.4556	0.3198	0.6234	0.5473	0.6448	0.6699	0.7025
	MUSIQ ↑	63.28	60.37	60.93	41.21	65.88	58.47	64.94	69.82	69.42
DRealSR	MANIQA ↑	0.3764	0.3733	0.3792	0.2461	0.4260	0.3836	0.4539	0.5406	0.5781
	TOPIQ ↑	0.5502	0.5147	0.5124	0.3207	0.5743	0.4883	0.5722	0.6887	0.6985
	PSNR ↑	28.74	28.62	28.17	29.72	28.03	28.42	26.67	27.89	26.82
	SSIM ↑	0.8033	0.8050	0.8126	0.8264	0.7523	0.7629	0.6548	0.7565	0.7298
	LPIPS ↓	0.2858	0.2818	0.2792	0.3099	0.3284	0.4036	0.4517	0.3273	0.3689
	CLIPQA ↑	0.5091	0.4507	0.4473	0.3813	0.6357	0.5286	0.6391	0.6708	0.6776
Real200	MUSIQ ↑	57.16	54.28	53.95	42.41	58.51	49.73	60.91	65.09	64.40
	MANIQA ↑	0.3424	0.3436	0.3444	0.2845	0.3867	0.3322	0.4486	0.5115	0.5214
	TOPIQ ↑	0.5058	0.4621	0.4518	0.3482	0.5320	0.4380	0.5819	0.6574	0.6618
	CLIPQA ↑	0.5910	0.5554	0.5508	0.5157	0.7272	0.6759	0.7170	0.7167	0.7437
	MUSIQ ↑	67.65	66.12	65.80	61.26	70.63	66.98	68.92	72.14	71.62
	MANIQA ↑	0.3882	0.3861	0.3921	0.3196	0.4838	0.4713	0.4869	0.5588	0.5794
	TOPIQ ↑	0.5966	0.5530	0.5478	0.4793	0.6517	0.6124	0.6235	0.7142	0.7176
	NFE ↓	-	-	-	-	200	15	50	50	5
# Parameters		16.70M	16.70M	16.70M	8.07M	1409.1M	173.9M	1716.7M	2283.7M	1716.6M
Latency/Image ↓		0.06s	0.08s	0.08s	0.04s	18.90s	1.12s	5.85s	7.24s	1.03s

Conclusion

- ❑ We present *DoSSR*, a diffusion-based super-resolution framework that significantly ***enhances both efficiency and performance*** by integrating a domain shift strategy with pretrained diffusion models.
- ❑ Empirical validation on diverse SR benchmarks confirms that DoSSR ***achieves a 5-7 times speed improvement*** over existing methods, setting a new state-of-the-art.

Thanks!

Code:



<https://github.com/QinpengCui/DoSSR>