ONE-STEP EFFECTIVE DIFFUSION NETWORK FOR REAL-WORLD IMAGE

Rongyuan Wu et al.
NeurIPS 2025

DeepShark Lab 학부연구생 박정규

OVERVIEW

- Problem
- OSEDiff model
- Methodology
- Key Idea
- Training Loss
- Experiments
- Conclusion
- Limitation
- Q&A

Problem

기존 학습된 text – to – image defussion 모델들은 Real-ISR 문제를 해결하기 위해 모든 픽셀들이 랜덤 noise에서 시작하고 Input의 LQ image를 참고하여 노이즈들을 조금씩 정리하며 HQ image를 생성하는 방식을 사용

여러 확산 과정을 거쳐야 하므로 계산 비용이 증가.
매번 확률적으로 결과가 다르게 나오는 문제 발생.
원본과 다르게 복원될 가능성이 존재.
현실 이미지에 대한 noise 복원에 취약.

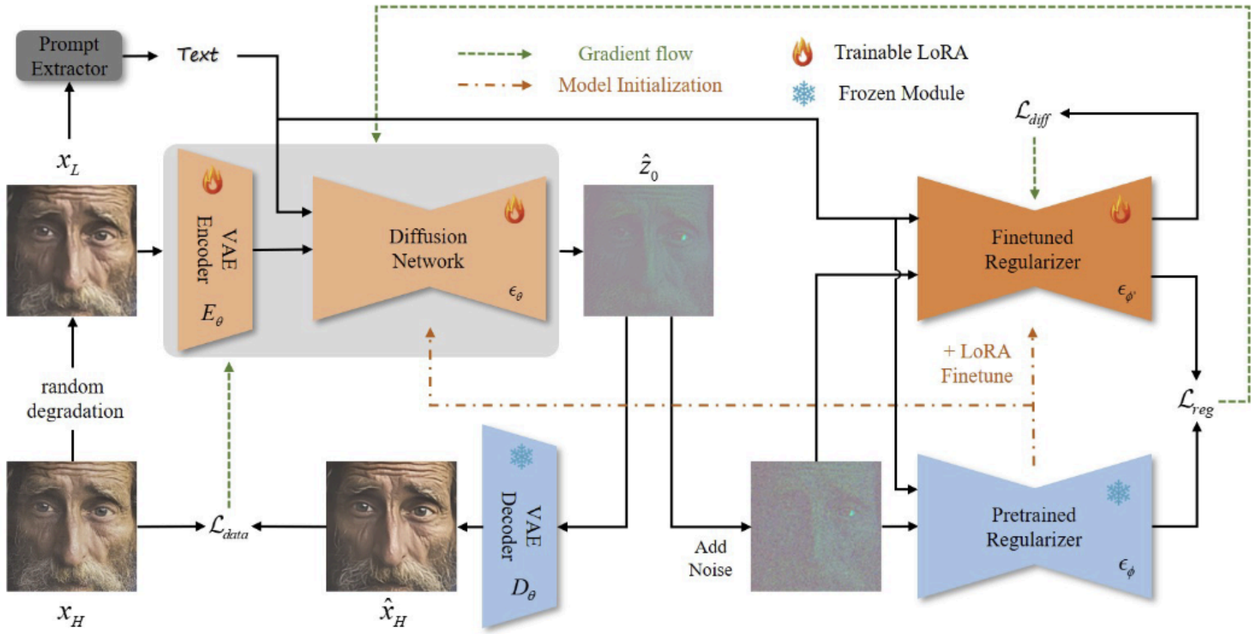
Noise 에 대한 학습 데이터를 HQ image에 가우시안 노이즈를 추가하여 LQ image로 변환 후 학습

OSEDiff model

기존 Real-ISR 문제를 해결하기 위한 방식의 확산과정 문제와 랜덤성 문제를 해결하기 위해 OSEDiff 모델 생성

기존 학습된 이미지 생성 모델을 이용하되, 학습 가능한 작은 레이어를 추가하고 Fine-tune 하여 현실 degradation에 적응 하도록 학습시킨 모델

Random noise를 시작으로 새로운 이미지를 생성하는게 아닌, input의 LQ image를 시작점으로 하여 한번의 확산 단계만으로 효율적이고 효과적인 HQ image 생성



Methodology

Real-ISR 문제를 수학적으로 모델링

$$\hat{\boldsymbol{x}}_H = \operatorname{argmin}_{\boldsymbol{x}_H} \left(\mathcal{L}_{data} \left(\Phi(\boldsymbol{x}_H), \boldsymbol{x}_L \right) + \lambda \mathcal{L}_{reg} \left(\boldsymbol{x}_H \right) \right)$$

$\Phi()$ Degration(열화)함수

L_{data} 입력된 저화질 이미지와 출력된 고화질 이미지를 다시 저화질로 바꾼 이미지간의 차이

λ 정규화 항의 가중치

L_{reg} 생성된 고화질 이미지가 얼마나 자연스러운지 확인

Methodology

Real-ISR 문제를 수학적으로 모델링

$$\theta^* = \text{argmin}_\theta \mathbb{E}_{(\boldsymbol{x}_L, \boldsymbol{x}_H)\sim S} \left[\mathcal{L}_{\text{data}} \left(G_\theta(\boldsymbol{x}_L), \boldsymbol{x}_H \right) + \lambda \mathcal{L}_{\text{reg}} \left(G_\theta(\boldsymbol{x}_L) \right) \right]$$

$$\mathcal{L}_{\text{reg}} = \mathcal{D}_{\text{KL}} \left(q_\theta(\hat{\boldsymbol{x}}_H) \| p(\boldsymbol{x}_H) \right)$$

KLdivergence : P 분포와 Q 분포가 있을때 , Q가 P를 얼마나 잘 따라가는지를 측정하는 모델

Key Idea

$$\hat{\boldsymbol{z}}_H = F_\theta(\boldsymbol{z}_L; c_y) \triangleq \frac{\boldsymbol{z}_L - \beta_T \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_L; T, c_y)}{\alpha_T},$$

$$\hat{\boldsymbol{x}}_H = G_\theta(\boldsymbol{x}_L) \triangleq D_\theta(F_\theta(E_\theta(\boldsymbol{x}_L); Y(\boldsymbol{x}_L))).$$

Encoder E_θ

VAE Encoder 사용

LQ image를 입력받으면 LQ-latent로 변환

Encoder을 학습 가능한 형태로 만들어 latent 벡터로 변환할 때 중요한 정보를 최대한 남기도록 구현

Finetuned Diffusion Network ϵ_θ

기존 Stable Diffusion모델인 U-Net사용하여 LQ latent를 HQ latent로 변환

LoRA를 이용하여 U-Net fine-tuning

Decoder D_θ

기존 Stable Diffusion 모델의 VAE Decoder 그대로 사용

Decoder은 학습 파라메터 없이 기존 모델 그대로 사용

Training Loss

Data Loss

$$\mathcal{L}_{\text{data}}\left(G_\theta(\boldsymbol{x}_L), \boldsymbol{x}_H\right) = \mathcal{L}_{\text{MSE}}\left(G_\theta(\boldsymbol{x}_L), \boldsymbol{x}_H\right) + \lambda_1 \mathcal{L}_{\text{LPIPS}}\left(G_\theta(\boldsymbol{x}_L), \boldsymbol{x}_H\right)$$

Regularization Loss

$$\mathcal{L}_{\text{reg}}\left(G_\theta(\boldsymbol{x}_L)\right) = \mathcal{L}_{\text{VSD}}\left(G_\theta(\boldsymbol{x}_L), c_y\right) = \mathcal{L}_{\text{VSD}}\left(G_\theta(\boldsymbol{x}_L), Y(\boldsymbol{x}_L)\right)$$

OSEDiff Loss

$$\mathcal{L}\left(G_\theta(\boldsymbol{x}_L), \boldsymbol{x}_H\right) = \mathcal{L}_{\text{data}}\left(G_\theta(\boldsymbol{x}_L), \boldsymbol{x}_H\right) + \lambda_2 \mathcal{L}_{\text{reg}}\left(G_\theta(\boldsymbol{x}_L)\right)$$

Experiments

Training Set

Real World image

LSDIR 데이터셋

FFHQ 데이터 셋

Test Set

DIV2K-Val set

Real-World image

Datasets	Methods	PSNR↑	SSIM↑	LPIPS↓	DISTS↓	FID↓	NIQE↓	MUSIQ↑	MANIQA↑	CLIPIQA↑
DIV2K-Val	StableSR-s200	23.26	0.5726	**0.3113**	0.2048	**24.44**	4.7581	65.92	0.6192	0.6771
	DiffBIR-s50	23.64	0.5647	0.3524	0.2128	30.72	**4.7042**	65.81	0.6210	0.6704
	SeeSR-s50	23.68	0.6043	0.3194	**0.1968**	**25.90**	4.8102	**68.67**	**0.6240**	**0.6936**
	PASD-s20	23.14	0.5505	0.3571	0.2207	29.20	**4.3617**	**68.95**	**0.6483**	**0.6788**
	ResShift-s15	**24.65**	**0.6181**	0.3349	0.2213	36.11	6.8212	61.09	0.5454	0.6071
	SinSR-s1	**24.41**	0.6018	0.3240	0.2066	35.57	6.0159	62.82	0.5386	0.6471
	OSEDiff-s1	23.72	**0.6108**	**0.2941**	**0.1976**	26.32	4.7097	67.97	0.6148	0.6683
DrealSR	StableSR-s200	28.03	0.7536	0.3284	**0.2269**	148.98	6.5239	58.51	0.5601	0.6356
	DiffBIR-s50	26.71	0.6571	0.4557	0.2748	166.79	**6.3124**	61.07	0.5930	0.6395
	SeeSR-s50	28.17	**0.7691**	**0.3189**	0.2315	**147.39**	6.3967	**64.93**	**0.6042**	0.6804
	PASD-s20	27.36	0.7073	0.3760	0.2531	156.13	**5.5474**	**64.87**	**0.6169**	**0.6808**
	ResShift-s15	**28.46**	0.7673	0.4006	0.2656	172.26	8.1249	50.60	0.4586	0.5342
	SinSR-s1	**28.36**	0.7515	0.3665	0.2485	170.57	6.9907	55.33	0.4884	0.6383
	OSEDiff-s1	27.92	**0.7835**	**0.2968**	**0.2165**	**135.30**	6.4902	64.65	0.5899	**0.6963**
RealSR	StableSR-s200	24.70	0.7085	0.3018	0.2288	128.51	5.9122	65.78	0.6221	0.6178
	DiffBIR-s50	24.75	0.6567	0.3636	0.2312	128.99	5.5346	64.98	0.6246	0.6463
	SeeSR-s50	25.18	0.7216	**0.3009**	**0.2223**	125.55	**5.4081**	**69.77**	**0.6442**	0.6612
	PASD-s20	25.21	0.6798	0.3380	0.2260	**124.29**	**5.4137**	68.75	**0.6487**	**0.6620**
	ResShift-s15	**26.31**	**0.7421**	0.3460	0.2498	141.71	7.2635	58.43	0.5285	0.5444
	SinSR-s1	**26.28**	**0.7347**	0.3188	0.2353	135.93	6.2872	60.80	0.5385	0.6122
	OSEDiff-s1	25.15	0.7341	**0.2921**	**0.2128**	**123.49**	5.6476	**69.09**	0.6326	**0.6693**

빨간색 : 가장 우수

파란색 : 두번째 우수

Experiments

Training Set

Real World image

DIV2K random crop image

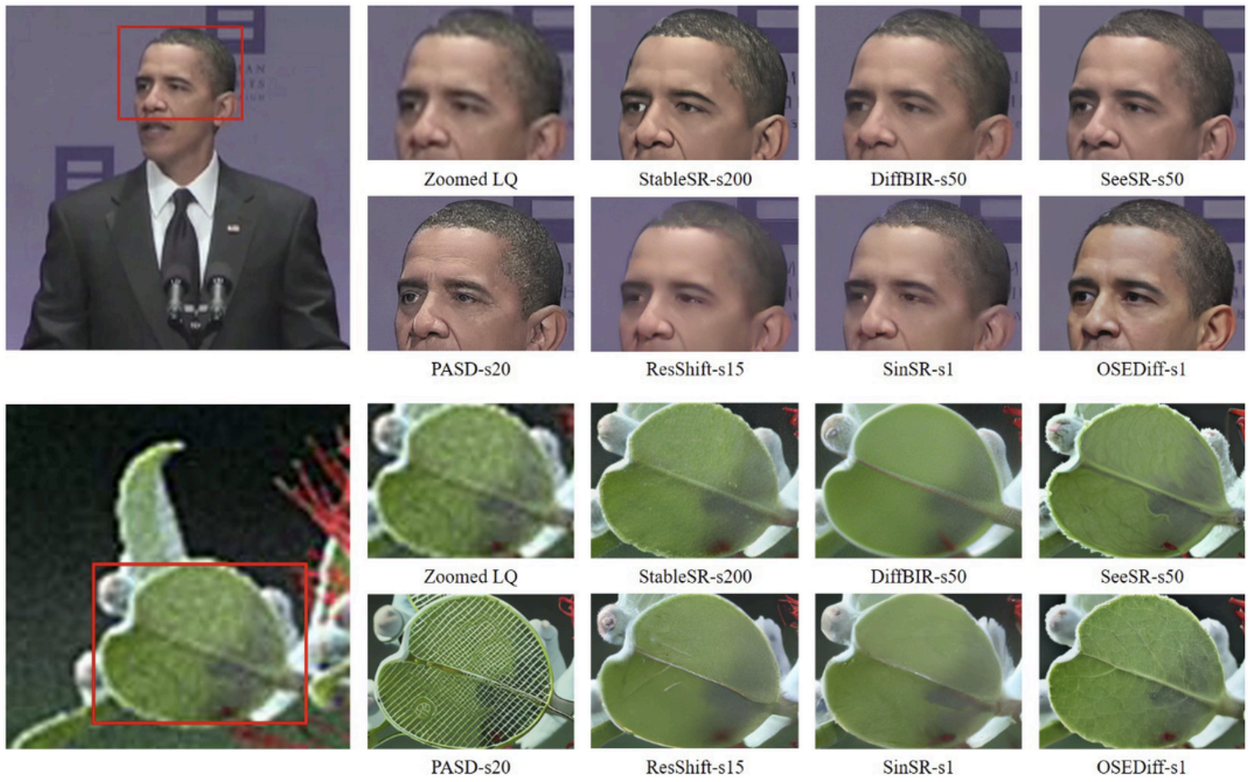
약 3000개의 HQ-LQ 쌍으로 구성

Test Set

DIV2K-Val set 100개

	StableSR	DiffBIR	SeeSR	PASD	ResShift	SinSR	OSEDiff
Inference Step	200	50	50	20	15	1	1
Inference Time (s)	11.50	2.72	4.30	2.80	0.71	0.13	0.11
MACs (G)	79940	24234	65857	29125	5491	2649	2265
# Total Param (M)	1410	1717	2524	1900	119	119	1775
# Trainable Param (M)	150.0	380.0	749.9	625.0	118.6	118.6	8.5

Experiments



☑ Zoomed image

다른 모델에 비해 자연스러운 디테일 복원 가능

☑ Real World Image

Prompt 참고하여 디테일 복원 가능

Conclusion & Limitation

Conclusion

OSEDiff 모델은 기존 다단계 확산 모델과 달리, LQ image를 시작점으로 직접 사용하여 무작위 노이즈와 관련된 불확실성을 제거하고, 학습 가능한 LoRA 레이어를 사용하여 기존에 학습된 확산 네트워크를 Fine Tuning 함으로써, 복잡한 실제 이미지 noise에 잘 적응할 수 있다.

Limitation

앞으로 OSEDiff 모델의 세부 정보 생성 능력을 더 발전 시킬 수 있다. 또한 작은 장면 텍스트와 같은 미세한 구조를 재구성하는데 한계가 있다.

Q&A

THANK YOU

For your attention