Image Restoration

2023년도 하계 세미나

Sogang University
Vision & Display Systems Lab, Dept. of Electronic Engineering

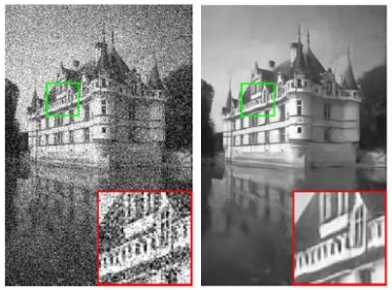
Presented By
Yoon Chan Nam

Outline

- Background
  - Image restoration
  - Vision transformer
- Efficient and Explicit Modelling of Image Hierarchies for Image Restoration
  - CVPR 2023
- Activating More Pixels in Image Super-Resolution Transformer
  - CVPR 2023

Background

- Image restoration



Denoising

Deblur

Deraining

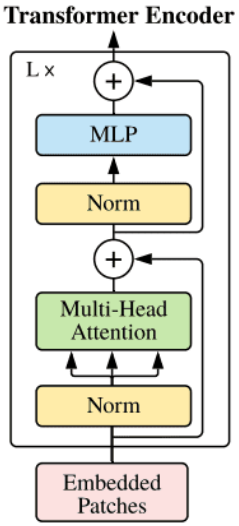
JPEG artifact removal

- Super resolution
  - Classic SR, Blind SR, Stereo SR etc..

Super resolution

Background

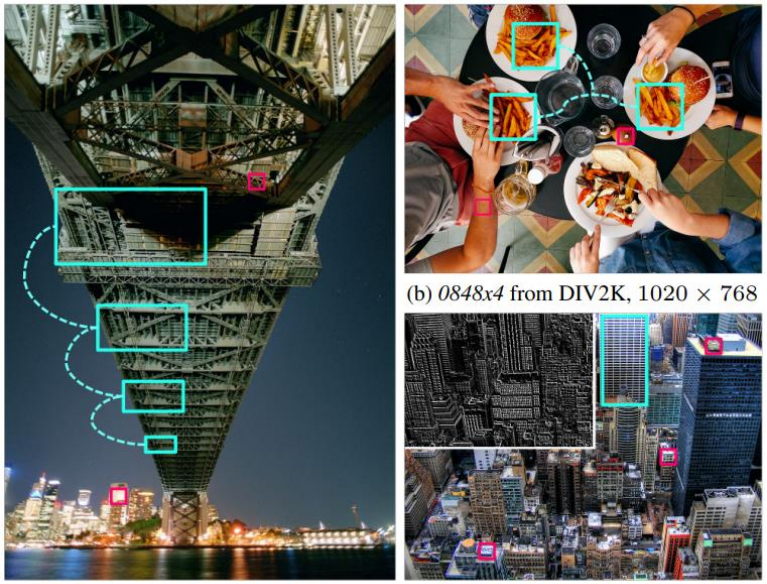
- Vision transformer
  - CNN은 long range dependency를 효과적으로 모델링 하기 어려운 단점을 해결
    - CNN kernel은 content-independent 하기 때문
  - 자연어 처리에서 사용되는 transformer를 vision task에 적용
    - 이미지를 patch 단위로 나누고 linear projection을 통해 patch embedding 생성
    - Patch embedding에 position embedding을 더해주고 transformer encoder의 입력으로 사용
    - Multi-head attention을 통한 patch embedding들 간의 관계를 global 하게 고려
    - 최종적으로 MLP를 통해 classification



- **Efficient and Explicit Modelling of Image Hierarchies for Image Restoration**
  - GRL
  - CVPR 2023

Introduction

- Image 구조를 효과적이고 명시적으로 모델링 하는 메커니즘을 제안
  - Image 의 두 가지 중요한 특성인 cross-scale similarity와 anisotropy를 고려
    - **G**lobal, **R**egional and **L**ocal의 다양한 Image 계층을 복원에 사용
      - GRL 이라는 새로운 네트워크 구조 제안
    - CNN으로 모델링 하기 어려운 global feature를 위한 새로운 attention 구조 제안
      - Anchored stripe self-attention을 제안하여 W-MSA, Channel attention과 함께 사용



(a) *bridge* from ICB, 2749 × 4049

(b) *0848x4* from DIV2K, 1020 × 768

(c) *073* from Urban100, 1024 × 765

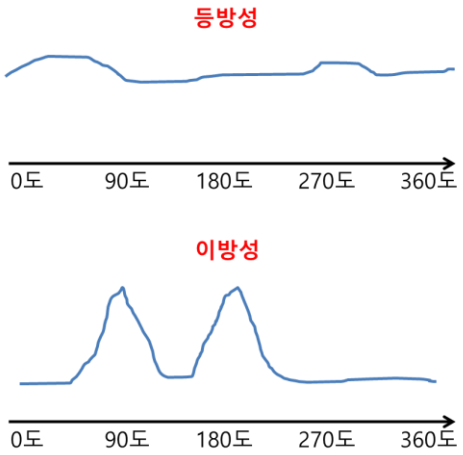
[하늘색 box는 global features를 의미하고, 분홍색 box는 local feature를 의미함]

Introduction

- Anisotropic (이방성)
  - 같은 방향을 갖는 성질
    - 영상처리 관점에서 특정 방향 성분을 많이 가지고 있을 수록 Anisotropic 특성이 높음
  - 반대되는 말로 isotropic (등방성)이 있음

Anisotropic (이방성)↑

Isotropic (등방성)↑



Motivation I: Self-attention for dependency modelling

- Window self-attention의 한계
  - 일반적으로 8×8 크기의 widow를 사용
    - 모델링의 용량을 local 또는 regional 범위로 제한 시키는 문제가 있음
    - Global self-attention으로 global feature를 모델링 할 수 있음

- Global self-attention의 한계
  - Image 의 전체 해상도를 입력으로 사용

$$\mathbf{Y} = \text{Softmax}\left(\mathbf{Q} \cdot \mathbf{K}^T / \sqrt{d}\right) \cdot \mathbf{V}$$

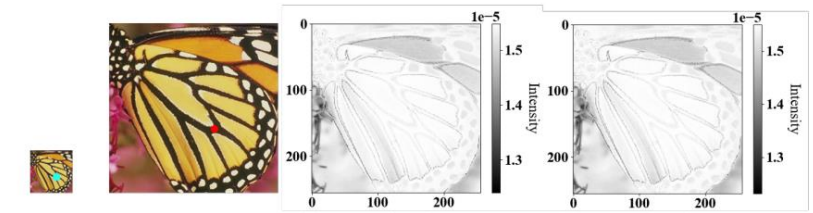
$$\mathbf{M} = \text{Softmax}(\mathbf{Q} \cdot \mathbf{K}^T / \sqrt{d})$$

- Attention map M을 계산할 때 pixel 수 N 에 대해서 연산량이 quadratic 하게 증가하는 문제가 있음

$$O(N^2)$$

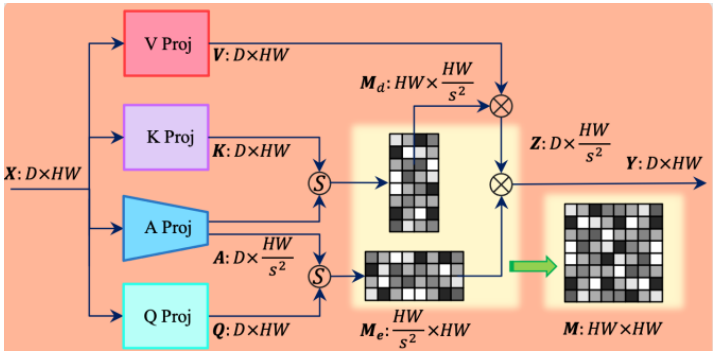
Motivation II: cross-scale similarity

- Anchored self-attention
  - Cross-scale similarity
    - Image의 scale에 상관없이 동일한 위치의 pixel에 대해서 attention map이 유사하게 출력됨



(a) LR image (b) HR image (c) Attention between the cyan and other pixels in (b). (d) Attention between the red pixel and other pixels in (b).

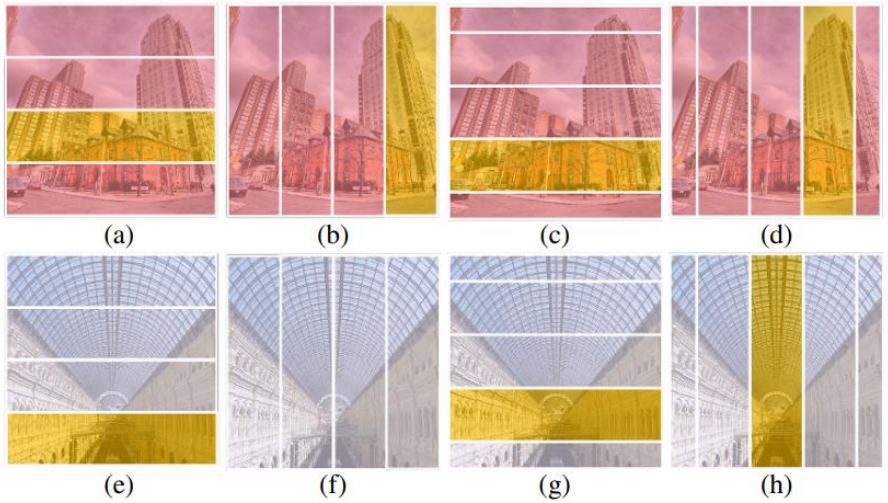
    - Cross-scale similarity에서 영감을 얻어 anchor **A**를 추가로 만들어 순차적으로 attention map을 구함
      - Anchor **A**는 image feature를 down-scaling 하여 요약한 image
      - 기존 global attention의 연산 량 $O(N^2)$에서 $O(NM)$ 으로 줄일 수 있음
        - ✓N: 원본 image의 pixel 수 ,M: anchor **A**의 pixel 수 ($M < N$)



$$\mathbf{Y} = \mathbf{M}_e \cdot \mathbf{Z} = \mathbf{M}_e \cdot (\mathbf{M}_d \cdot \mathbf{V})$$
$$\mathbf{M}_d = \mathrm{Softmax}(\mathbf{A} \cdot \mathbf{K}^T / \sqrt{d})$$
$$\mathbf{M}_e = \mathrm{Softmax}(\mathbf{Q} \cdot \mathbf{A}^T / \sqrt{d})$$

Motivation III: anisotropic image features

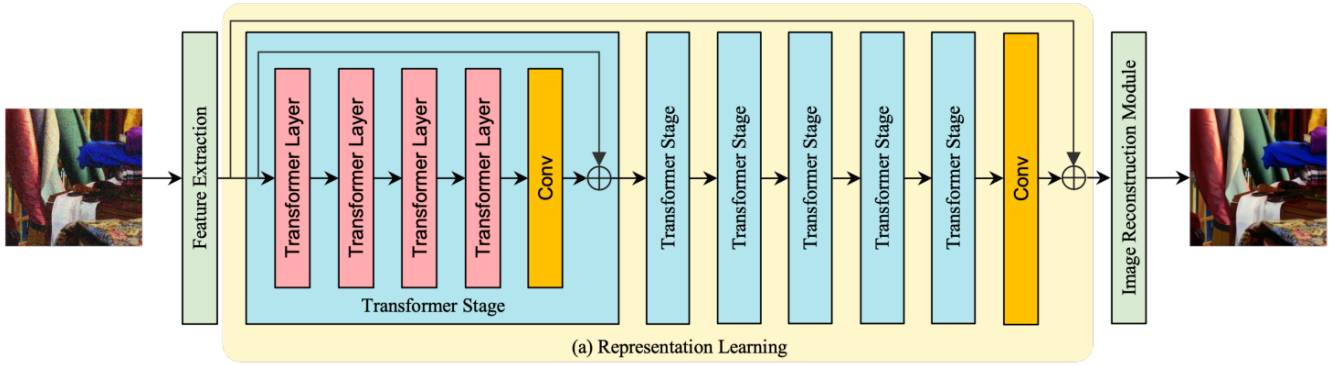
- Stripe attention mechanism
  - Anisotropic image features
    - 자연 image에는 보통 anisotropic한 특징이 있음
      - ⁘ 이에 대한 특징을 포착하고 모델링 하기 위해 stripe attention 기법을 도입
  - 네 가지 stripe mode를 구성하여 stripe 단위로 anchored self-attention을 수행
    - Horizontal, vertical, shifted horizontal and shifted vertical
      - ⁘ Global self-attention의 모델링 용량을 유지하면서, 계산 복잡도를 낮출 수 있음



[Natural image의 anisotropic한 특징과, 네 가지 stripe mode]

Methods

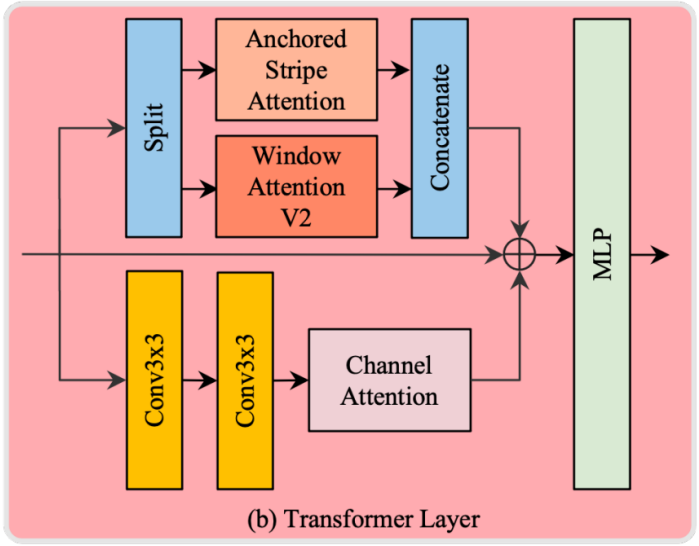
- Network architecture
  - Feature extraction layer
    - 간단한 convolution layer로 구성, image feature를 추출함
  - Representation learning
    - 추출된 feature를 더욱 표현력이 좋은 feature로 만듦
    - Transformer stage는 여러 Transformer layer로 구성되며, convolution layer로 마무리됨
  - Image reconstruction module
    - 이전 작업에 의해 계산된 feature로 clean한 image 복원



[전체 network architecture]

Methods

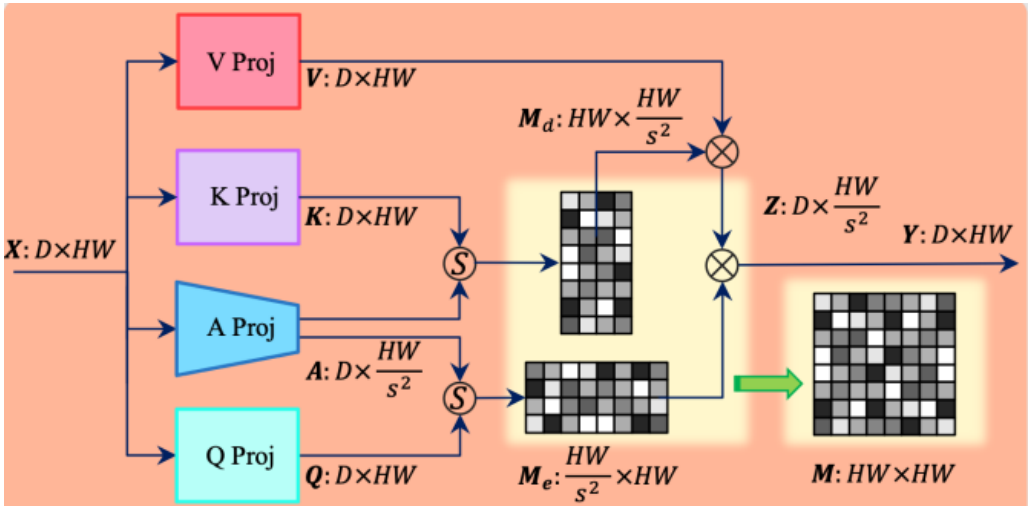
- Transformer layer (Efficient Mixed Attention Transformer Block)
  - Channel attention
    - 앞 단에 3x3 convolution을 사용하여 local 범위의 구조를 capture함
  - Anchored stripe-attention & Window attention V2
    - Regional 범위의 구조를 capture 하기 위해 Swin Transformer V2의 window self-attention을 사용
    - Global 범위의 구조를 capture 하기 위해 제안한 Anchored stripe self-attention을 사용



[Transformer layer의 구조]

Methods

- Anchored stripe self-attention
  - **Q, K, V** 이외에 anchor **A**를 추가하여 두개의 attention map M_d, M_e을 생성
    - Down-sampling 계수 s 를 사용하여 Anchor A를 생성
      - Cross-scale similarity에 근거하여 anchor 사용
      - s = 2
    - Attention map M_d, M_e은 기존 global attention map M과 유사한 역할을 하지만 계산 복잡도가 적음
      - 기존 global attention의 연산 량 $O(N^2)$에서 $O(NM)$ 으로 줄임
        - ✓N: 원본 image의 pixel 수 ,M: anchor **A**의 pixel 수 ($M < N$)



[Anchored stripe self-attention 의 구조]

Experiments results

Table 1. ***Defocus deblurring*** results. **S:** single-image defocus deblurring. **D:** dual-pixel defocus deblurring.

Method	Indoor Scenes				Outdoor Scenes				Combined			
	PSNR↑	SSIM↑	MAE↓	LPIPS↓	PSNR↑	SSIM↑	MAE↓	LPIPS↓	PSNR↑	SSIM↑	MAE↓	LPIPS↓
EBDB$_S$ [30]	25.77	0.772	0.040	0.297	21.25	0.599	0.058	0.373	23.45	0.683	0.049	0.336
DMENet$_S$ [40]	25.50	0.788	0.038	0.298	21.43	0.644	0.063	0.397	23.41	0.714	0.051	0.349
JNB$_S$ [60]	26.73	0.828	0.031	0.273	21.10	0.608	0.064	0.355	23.84	0.715	0.048	0.315
DPDNet$_S$ [1]	26.54	0.816	0.031	0.239	22.25	0.682	0.056	0.313	24.34	0.747	0.044	0.277
KPAC$_S$ [62]	27.97	0.852	0.026	0.182	22.62	0.701	0.053	0.269	25.22	0.774	0.040	0.227
IFAN$_S$ [41]	28.11	0.861	0.026	0.179	22.76	0.720	0.052	0.254	25.37	0.789	0.039	0.217
Restormer$_S$ [76]	28.87	0.882	0.025	0.145	23.24	0.743	0.050	0.209	25.98	0.811	0.038	0.178
GRL$_S$-B	29.06	0.886	0.024	0.139	23.45	0.761	0.049	0.196	26.18	0.822	0.037	0.168
DPDNet$_D$ [1]	27.48	0.849	0.029	0.189	22.90	0.726	0.052	0.255	25.13	0.786	0.041	0.223
RDPD$_D$ [2]	28.10	0.843	0.027	0.210	22.82	0.704	0.053	0.298	25.39	0.772	0.040	0.255
Uformer$_D$ [74]	28.23	0.860	0.026	0.199	23.10	0.728	0.051	0.285	25.65	0.795	0.039	0.243
IFAN$_D$ [41]	28.66	0.868	0.025	0.172	23.46	0.743	0.049	0.240	25.99	0.804	0.037	0.207
Restormer$_D$ [76]	29.48	0.895	0.023	0.134	23.97	0.773	0.047	0.175	26.66	0.833	0.035	0.155
GRL$_D$-B	29.83	0.903	0.022	0.114	24.39	0.795	0.045	0.150	27.04	0.847	0.034	0.133

Table 2. ***Single-image motion deblurring*** results. GoPro dataset [51] is used for training.

Method	GoPro [51] PSNR↑ / SSIM↑	HIDE [59] PSNR↑ / SSIM↑	Average PSNR↑ / SSIM↑
DeblurGAN [37]	28.70 / 0.858	24.51 / 0.871	26.61 / 0.865
Nah *et al.* [51]	29.08 / 0.914	25.73 / 0.874	27.41 / 0.894
DeblurGAN-v2 [38]	29.55 / 0.934	26.61 / 0.875	28.08 / 0.905
SRN [64]	30.26 / 0.934	28.36 / 0.915	29.31 / 0.925
Gao *et al.* [20]	30.90 / 0.935	29.11 / 0.913	30.01 / 0.924
DBGAN [81]	31.10 / 0.942	28.94 / 0.915	30.02 / 0.929
MT-RNN [53]	31.15 / 0.945	29.15 / 0.918	30.15 / 0.932
DMPHN [79]	31.20 / 0.940	29.09 / 0.924	30.15 / 0.932
Suin *et al.* [63]	31.85 / 0.948	29.98 / 0.930	30.92 / 0.939
SPAIR [55]	32.06 / 0.953	30.29 / 0.931	31.18 / 0.942
MIMO-UNet+ [8]	32.45 / 0.957	29.99 / 0.930	31.22 / 0.944
IPT [5]	32.52 / -	- / -	- / -
MPRNet [77]	32.66 / 0.959	30.96 / 0.939	31.81 / 0.949
Restormer [76]	32.92 / 0.961	31.22 / 0.942	32.07 / 0.952
GRL-B (ours)	33.93 / 0.968	31.65 / 0.947	32.79 / 0.958

Table 3. ***Single-image motion deblurring*** results on RealBlur [57] dataset. The networks are trained and tested on RealBlur dataset.

Method	RealBlur-R [57] PSNR↑ / SSIM↑	RealBlur-J [57] PSNR↑ / SSIM↑	Average PSNR↑ / SSIM↑
DeblurGAN-v2 [38]	36.44 / 0.935	29.69 / 0.870	33.07 / 0.903
SRN [64]	38.65 / 0.965	31.38 / 0.909	35.02 / 0.937
MPRNet [77]	39.31 / 0.972	31.76 / 0.922	35.54 / 0.947
MIMO-UNet+ [8]	- / -	32.05 / 0.921	- / -
MAXIM-3S [67]	39.45 / 0.962	32.84 / 0.935	36.15 / 0.949
BANet [66]	39.55 / 0.971	32.00 / 0.923	35.78 / 0.947
MSSNet [34]	39.76 / 0.972	32.10 / 0.928	35.93 / 0.950
Stripformer [65]	39.84 / 0.974	32.48 / 0.929	36.16 / 0.952
GRL-B (ours)	40.20 / 0.974	32.82 / 0.932	36.51 / 0.953

Table 4. ***Color and grayscale image denoising*** results. Model complexity and prediction accuracy are shown for better comparison.

Method	# Params [M]	Color									Grayscale								
		CBSD68 [49]			Kodak24 [16]			McMaster [83]			Urban100 [28]			Set12 [82]			BSD68 [49]			Urban100 [28]		
		σ=15	σ=25	σ=50	σ=15	σ=25	σ=50	σ=15	σ=25	σ=50	σ=15	σ=25	σ=50	σ=15	σ=25	σ=50	σ=15	σ=25	σ=50	σ=15	σ=25	σ=50
DnCNN [32]	0.56	33.90	31.24	27.95	34.60	32.14	28.95	33.45	31.52	28.62	32.98	30.81	27.59	32.86	30.44	27.18	31.73	29.23	26.23	32.64	29.95	26.26
RNAN [85]	8.96	-	-	28.27	-	-	29.58	-	-	29.72	-	-	29.08	-	-	27.70	-	-	26.48	-	-	27.65
IPT [5]	115.33	-	-	28.39	-	-	29.64	-	-	29.98	-	-	29.71	-	-	-	-	-	-	-	-	-
EDT-B [42]	11.48	34.39	31.76	28.56	35.37	32.94	29.87	35.61	33.34	30.25	35.22	33.07	30.16	-	-	-	-	-	-	-	-	-
DRUNet [80]	32.64	34.30	31.69	28.51	35.31	32.89	29.86	35.40	33.14	30.08	34.81	32.60	29.61	33.25	30.94	27.90	31.91	29.48	26.59	33.44	31.11	27.96
SwinIR [43]	11.75	34.42	31.78	28.56	35.34	32.89	29.79	35.61	33.20	30.22	35.13	32.90	29.82	33.36	31.01	27.91	31.97	29.50	26.58	33.70	31.30	27.98
Restormer [76]	26.13	34.40	31.79	28.60	35.47	33.04	30.01	35.61	33.34	30.30	35.13	32.96	30.02	33.42	31.08	28.00	31.96	29.52	26.62	33.79	31.46	28.29
GRL-T	0.88	34.30	31.66	28.45	35.24	32.78	29.67	35.49	33.18	30.06	35.08	32.84	29.78	33.29	30.92	27.78	31.90	29.43	26.49	33.66	31.23	27.89
GRL-S	3.12	34.36	31.72	28.51	35.32	32.88	29.77	35.59	33.29	30.18	35.24	33.07	30.09	33.36	31.02	27.91	31.93	29.47	26.54	33.84	31.49	28.24
GRL-B	19.81	34.45	31.82	28.62	35.43	33.02	29.93	35.73	33.46	30.36	35.54	33.35	30.46	33.47	31.12	28.03	32.00	29.54	26.60	34.09	31.80	28.59

Table 5. ***Classical image SR*** results. Results of both lightweight models and accurate models are summarized.

Method	Scale	# Params [M]	Set5 [3]		Set14 [78]		BSD100 [49]		Urban100 [28]		Manga109 [50]	
			PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
RCAN [84]	×2	15.44	38.27	0.9614	34.12	0.9216	32.41	0.9027	33.34	0.9384	39.44	0.9786
SAN [11]	×2	15.71	38.31	0.9620	34.07	0.9213	32.42	0.9028	33.10	0.9370	39.32	0.9792
HAN [52]	×2	63.61	38.27	0.9614	34.16	0.9217	32.41	0.9027	33.35	0.9385	39.46	0.9785
IPT [5]	×2	115.48	38.37	-	34.43	-	32.48	-	33.76	-	-	-
SwinIR [43]	×2	0.88	38.14	0.9611	33.86	0.9206	32.31	0.9012	32.76	0.9340	39.12	0.9783
SwinIR [43]	×2	11.75	38.42	0.9623	34.46	0.9250	32.53	0.9041	33.81	0.9427	39.92	0.9797
EDT [42]	×2	0.92	38.23	0.9615	33.99	0.9209	32.37	0.9021	32.98	0.9362	39.45	0.9789
EDT [42]	×2	11.48	38.63	0.9632	34.80	0.9273	32.62	0.9052	34.27	0.9456	40.37	0.9811
GRL-T (ours)	×2	0.89	38.27	0.9627	34.21	0.9258	32.42	0.9056	33.60	0.9411	39.61	0.9790
GRL-S (ours)	×2	3.34	38.37	0.9632	34.64	0.9280	32.52	0.9069	34.36	0.9463	39.84	0.9793
GRL-B (ours)	×2	20.05	38.67	0.9647	35.08	0.9303	32.67	0.9087	35.06	0.9505	40.67	0.9818
RCAN [84]	×4	15.59	32.63	0.9002	28.87	0.7889	27.77	0.7436	26.82	0.8087	31.22	0.9173
SAN [11]	×4	15.86	32.64	0.9003	28.92	0.7888	27.78	0.7436	26.79	0.8068	31.18	0.9169
HAN [52]	×4	64.20	32.64	0.9002	28.90	0.7890	27.80	0.7442	26.85	0.8094	31.42	0.9177
IPT [5]	×4	115.63	32.64	-	29.01	-	27.82	-	27.26	-	-	-
SwinIR [43]	×4	0.90	32.44	0.8976	28.77	0.7858	27.69	0.7406	26.47	0.7980	30.92	0.9151
SwinIR [43]	×4	11.90	32.92	0.9044	29.09	0.7950	27.92	0.7489	27.45	0.8254	32.03	0.9260
EDT [42]	×4	0.92	32.53	0.8991	28.88	0.7882	27.76	0.7433	26.71	0.8051	31.35	0.9180
EDT [42]	×4	11.63	33.06	0.9055	29.23	0.7971	27.99	0.7510	27.75	0.8317	32.39	0.9283
GRL-T (ours)	×4	0.91	32.56	0.9029	28.93	0.7961	27.77	0.7523	27.15	0.8185	31.57	0.9219
GRL-S (ours)	×4	3.49	32.76	0.9058	29.10	0.8007	27.90	0.7568	27.90	0.8357	32.11	0.9267
GRL-B (ours)	×4	20.20	33.10	0.9094	29.37	0.8058	28.01	0.7611	28.53	0.8504	32.77	0.9325

Table 6. ***Grayscale image JPEG compression artifact removal*** results. As a comparison metric, the parameter count of FBCNN [29] GRL-S are 71.92M and 3.12M.

Set	QF	JPEG PSNR	JPEG SSIM	DnCNN [82] PSNR	DnCNN [82] SSIM	DCSC [18] PSNR	DCSC [18] SSIM	QGAC [14] PSNR	QGAC [14] SSIM	MWCNN [46] PSNR	MWCNN [46] SSIM	FBCNN [29] PSNR	FBCNN [29] SSIM	GRL-S PSNR	GRL-S SSIM
Classic5 [15]	10	27.82	0.760	29.40	0.803	29.62	0.810	29.84	0.812	30.01	0.820	30.12	0.822	30.20	0.829
	20	30.12	0.834	31.63	0.861	31.81	0.864	31.98	0.869	32.16	0.870	32.31	0.872	32.49	0.878
	30	31.48	0.867	32.91	0.886	33.06	0.888	33.22	0.892	33.43	0.893	33.54	0.894	33.72	0.899
	40	32.43	0.885	33.77	0.900	33.87	0.902	34.05	0.905	34.27	0.906	34.35	0.907	34.53	0.911
BSD500 [49]	10	27.80	0.768	29.21	0.809	29.32	0.813	29.46	0.821	29.61	0.820	29.67	0.821	29.74	0.823
	20	30.05	0.849	31.53	0.878	31.63	0.880	31.73	0.884	31.92	0.885	32.00	0.885	32.05	0.885
	30	31.37	0.884	32.90	0.907	32.99	0.908	33.07	0.912	33.30	0.912	33.37	0.913	33.43	0.912
	40	32.30	0.903	33.85	0.923	33.92	0.924	34.01	0.927	34.27	0.928	34.33	0.928	34.38	0.928

Table 7. ***Color image JPEG compression artifact removal*** *results.*

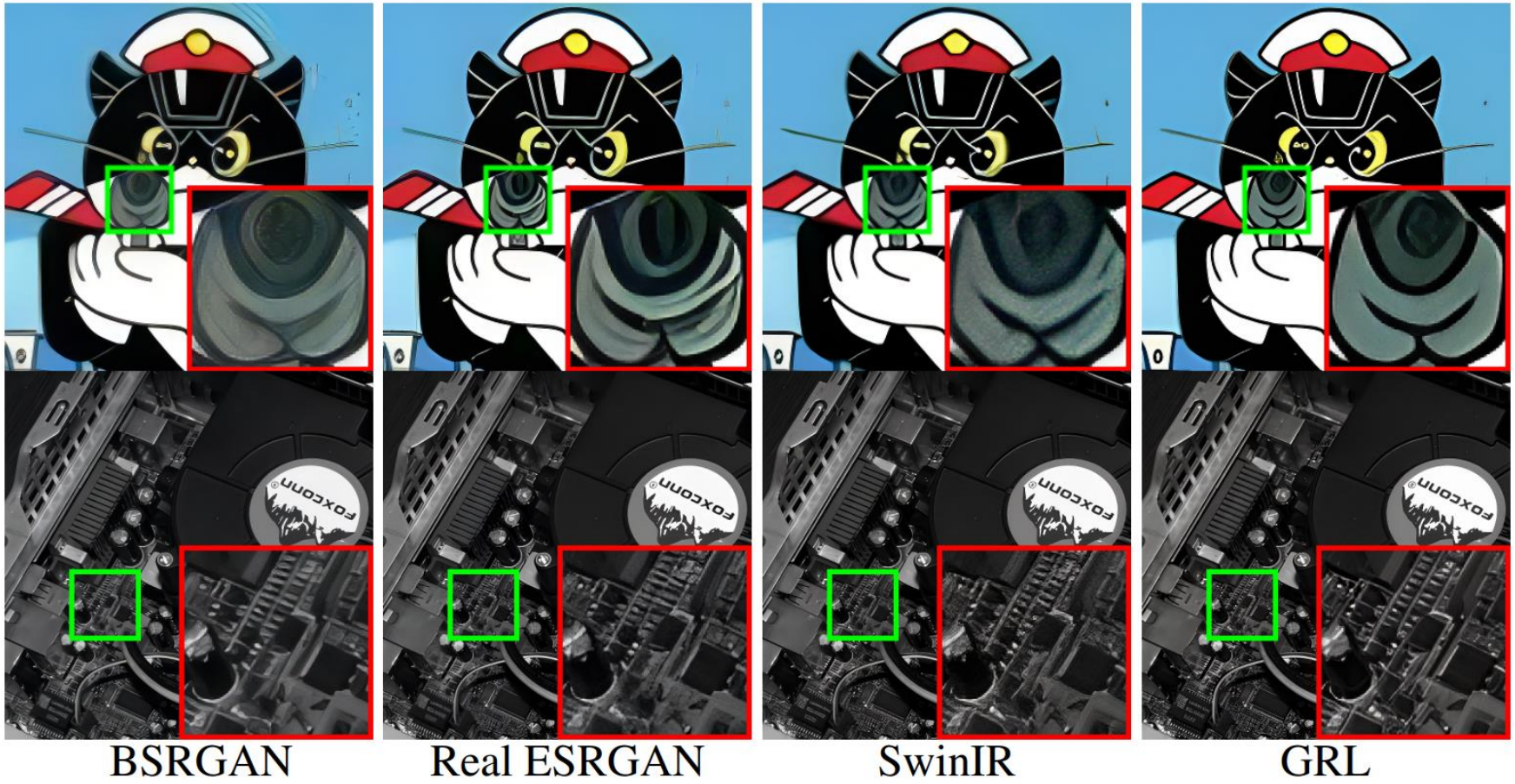
Set	QF	JPEG PSNR	JPEG SSIM	QGAC [14] PSNR	QGAC [14] SSIM	FBCNN [29] PSNR	FBCNN [29] SSIM	GRL-S PSNR	GRL-S SSIM
LIVE1 [58]	10	25.69	0.743	27.62	0.804	27.77	0.803	28.13	0.814
	20	28.06	0.826	29.88	0.868	30.11	0.868	30.49	0.878
	30	29.37	0.861	31.17	0.896	31.43	0.897	31.85	0.905
	40	30.28	0.882	32.05	0.912	32.34	0.913	32.79	0.920
BSD500 [49]	10	25.84	0.741	27.74	0.802	27.85	0.799	28.26	0.808
	20	28.21	0.827	30.01	0.869	30.14	0.867	30.57	0.875
	30	29.57	0.865	31.33	0.898	31.45	0.897	31.92	0.903
	40	30.52	0.887	32.25	0.915	32.36	0.913	32.86	0.919

Table 8. ***Image demosaicking*** results.

Datasets	Matlab	DDR [75]	DeepJoint [21]	MMNet [36]	RLDD [24]	RNAN [85]	DRUNet [80]	GRL-S (ours)
Kodak [16]	35.78	41.11	42.00	40.19	42.49	43.16	42.68	43.57
McMaster [83]	34.43	37.12	39.14	37.09	39.25	39.70	39.39	40.22

Experiments results

- Real world super resolution



Conclusions

- Image restoration을 위한 GRL network 제안
  - Anchored stripe self-attention module 제안
    - Image 의 cross-scale similarity와 anisotropic 특징에서 영감을 얻음
  - 다목적 image restoration을 위한 새로운 GRL network 구조 제안
    - Global, regional, local 영역을 모두 모델링 할 수 있음
  - 제안된 network는 다양한 image 복원 작업에 대한 최첨단 성능을 달성함

- **Activating More Pixels in Image Super-Resolution Transformer**
  - HAT
  - CVPR 2023

Introduction

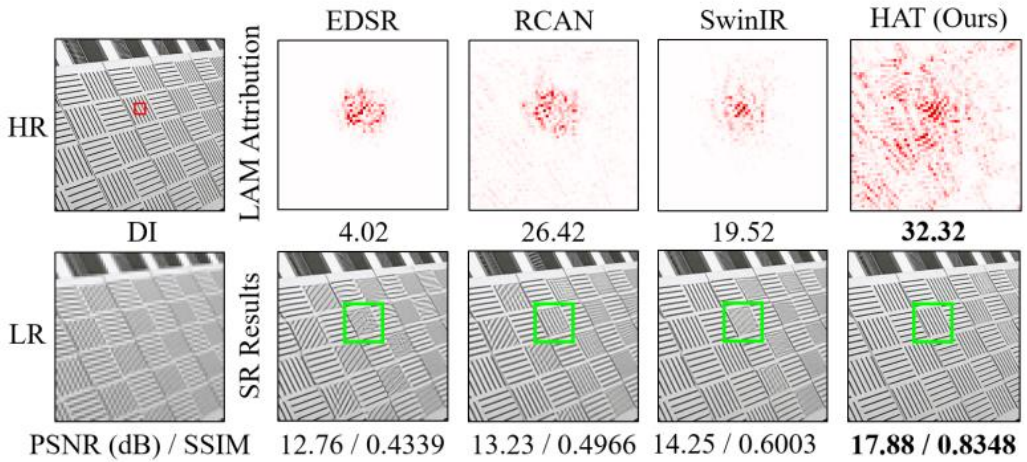
- Low level vision에서의 Transformer
  - 높은 수준의 성능 향상을 보여줌
  - 하지만 입력 image에 대해 제한된 spatial range만 활용 가능한 문제가 있음

- Hybrid Attention Transformer (HAT)를 제안
  - 기존 Transformer 구조를 개선하여 더 많은 pixel을 활성화
    - Channel attention과 window self-attention을 결합하여 사용
      - Global feature 와 local feature 모두 활용 하기 위함
    - Overlapping cross-attention module 제안
      - Cross window 정보를 더 잘 집계하기 위함

- 네트워크 성능 향상을 위한 same-task pre-training 진행

Motivation

- SwinIR의 한계
  - LAM (Local Attribution Map)
    - 어떤 픽셀이 선택한 영역을 복원하는데 가장 많이 기여하는지 시각화 하는 방법론

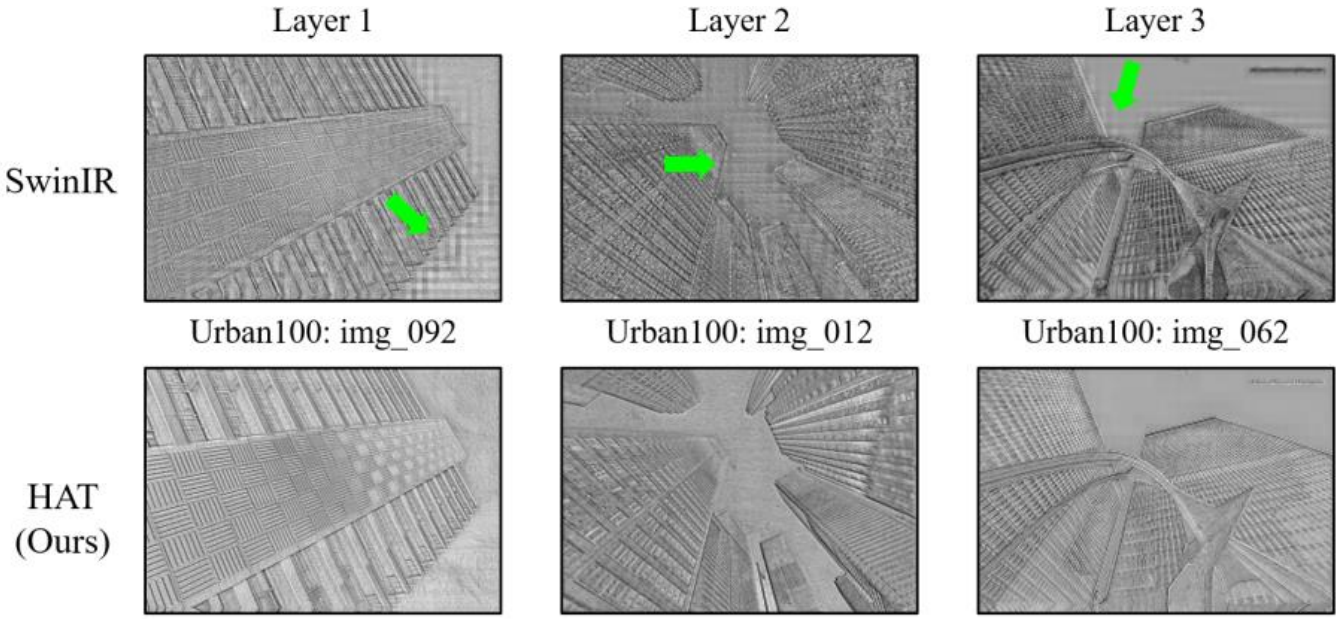


[각 모델 별 LAM 결과]

- EDSR, RCAN 과 같은 CNN 기반 방법보다 더 적은 영역을 참조함
  - SwinIR이 CNN 기반 방법보다 성능이 우수한 것과 모순됨
    - ✓SwinIR이 CNN 기반 방법보다 적은 pixel로도 mapping을 더 잘하며
    - ✓더 많은 pixel을 활용한다면 더 나은 성능을 얻을 수 있다고 볼 수 있음
- 더 많은 pixel을 activating 하기 위해 HAT를 제안

Motivation

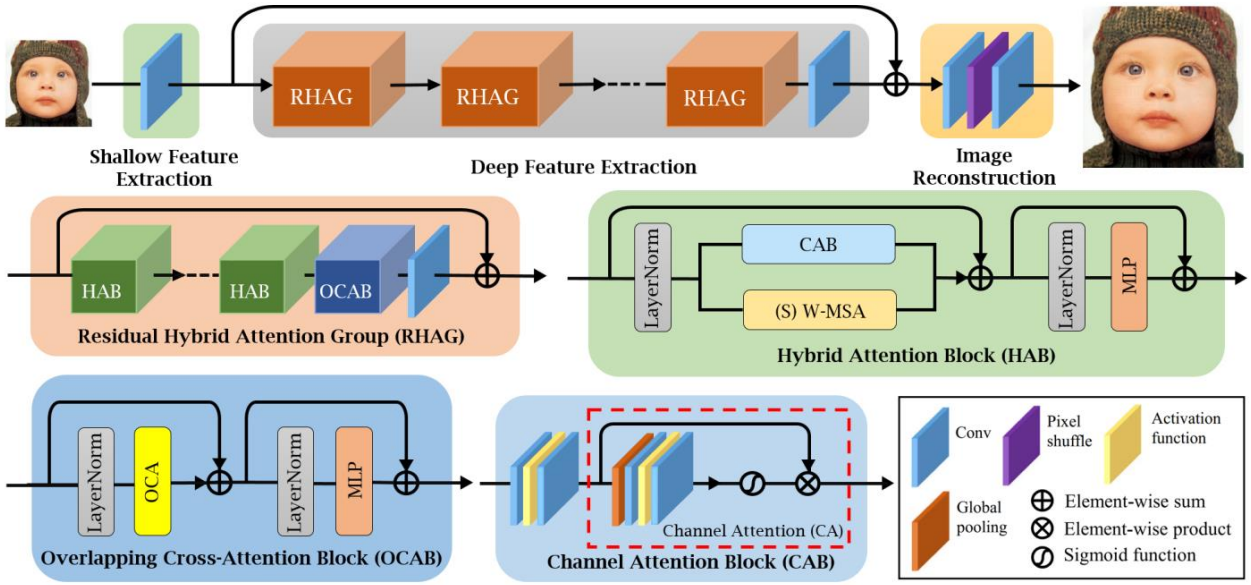
- SwinIR의 한계
  - Blocking artifact
    - SwinIR의 window partition mechanism에 의해 발생
    - 또한 Shifted window 방식이 cross window 방식을 구축하는데 비 효율적임
      - HAT에서 cross-window information interaction을 향상시켜 blocking artifact 완화



[Layer 마다의 중간 feature 출력 결과]

Method

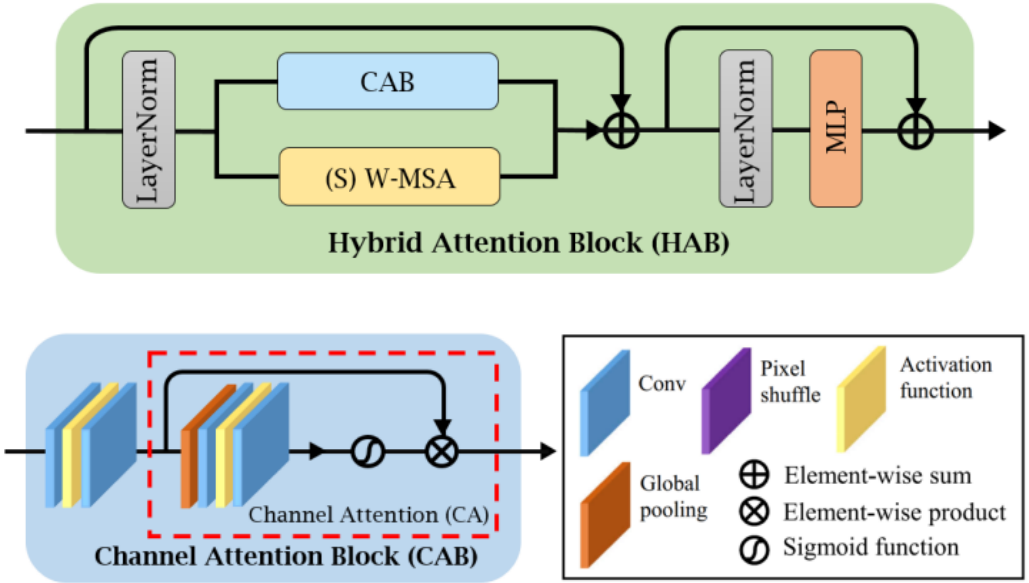
- Overall architecture
  - Shallow Feature Extraction
    - Simple convolution layer로 구성되며 shallow feature 추출
  - Deep Feature Extraction
    - RHAG(Residual Hybrid Attention Group) block으로 이루어진 deep feature 추출 구간
  - Image Reconstruction
    - Global residual 연결을 이용하여 shallow feature와 deep feature를 융합하여 복원



[Overall architecture of HAT network]

Method

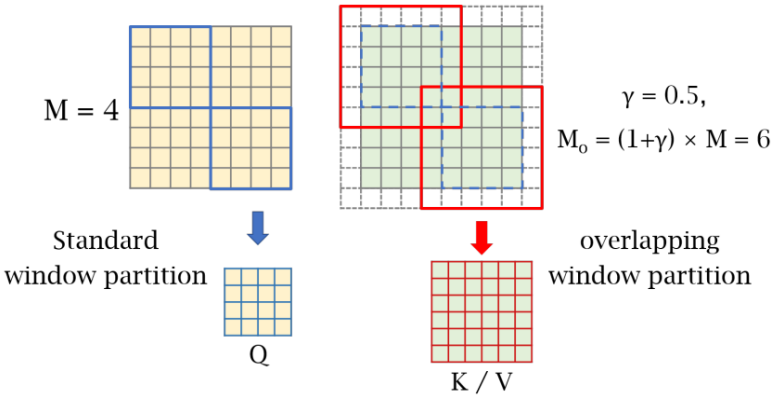
- HAB (Hybrid Attention Block)
  - SW-MSA (Shifted Window-based Multi-head Self-Attention)
    - SwinIR에서 사용하는 SW-MSA를 그대로 사용
  - CAB (Channel Attention Block)
    - Global 정보를 포함 할 수 있기 때문에 더 많은 pixel을 활성화 시킬 수 있음
    - CAB와 MSA의 충돌을 방지하기 위해 작은 상수 α를 CAB 출력에 곱해 줌
    - GELU activation 사용



[HAB (Hybrid Attention Block)의 구조]

Method

- OCAB (Overlapping Cross-Attention Block)
  - Overlapping 되는 window partition을 Key, Value로 사용
    - Query 는 standard window partition을 사용
      - 기존의 W-MSA는 Q, K, V를 모두 같은 window partition feature에서 사용
    - Standard window partition에서 γ값에 따라 overlapping window 크기 조절
      - 크기 일관성을 위해 zero padding을 사용
  - W-MSA로 인해 생기는 blocking artifact를 완화할 수 있음



[OCA (Overlapping Cross Attention)의 window partition 구성 예시]

Experiments results

- Pre-training strategy
  - Transformer의 성능 향상을 위해 대규모의 ImageNet에서 ×4 SR 모델을 pre-train함
    - 대용량 학습에 특화된 Transformer의 잠재력을 끌어올리기 위함
  - 이후에 각 특정 task에 맞는 dataset을 사용하여 fine-tunning
    - Same-task pre-training

- Ablation study

Table 2. Ablation study on the proposed OCAB and CAB.

	Baseline			
OCAB	X	✓	X	✓
CAB	X	X	✓	✓
PSNR	27.81dB	27.91dB	27.91dB	27.97dB

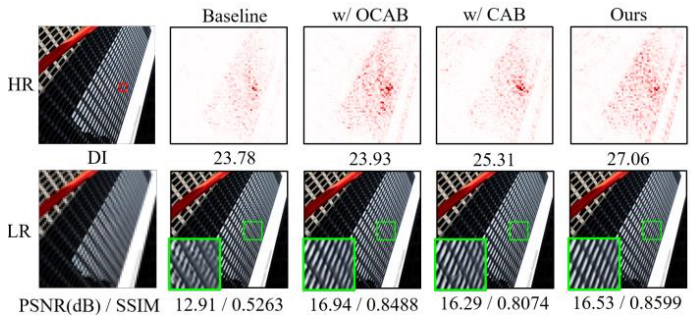


Table 3. Effects of the channel attention (CA) module in CAB.

Structure	w/o CA	w/ CA
PSNR / SSIM	27.92dB / 0.8362	27.97dB / 0.8367

Table 4. Effects of the weighting factor α in CAB.

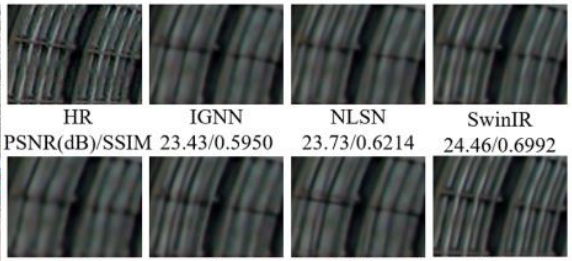
α	0	1	0.1	0.01
PSNR	27.81dB	27.86dB	27.90dB	27.97dB

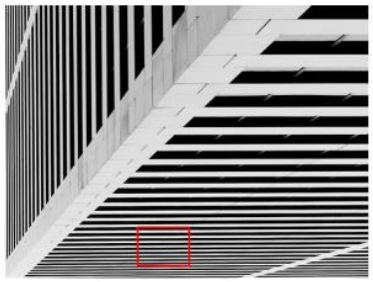
Table 5. Ablation study on the overlapping ratio of OCAB.

γ	0	0.25	0.5	0.75
PSNR	27.85dB	27.81dB	27.91dB	27.86dB

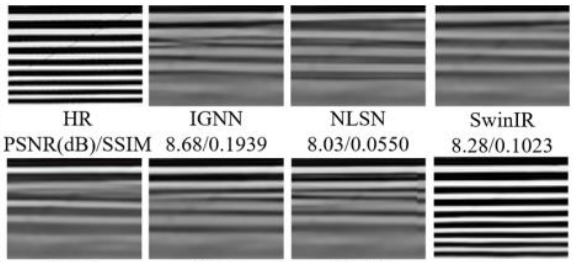
Experiments results

Urban100 (×4): img_002

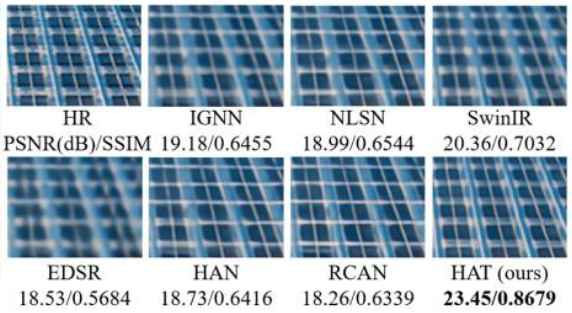




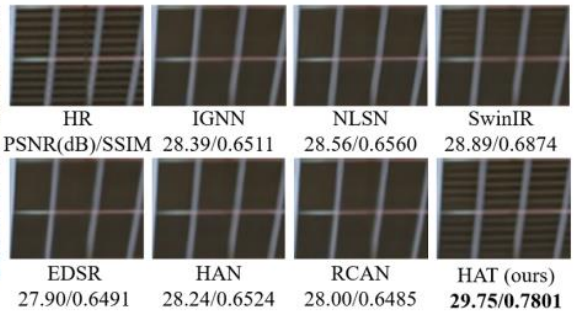
Urban100 (×4): img_011



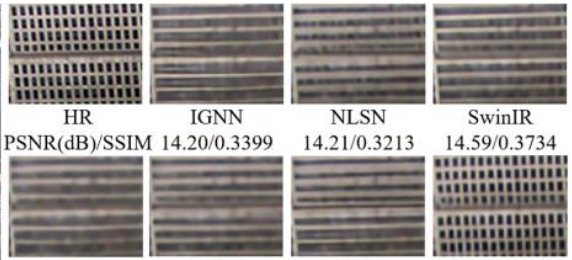
Urban100 (×4): img_030



Urban100 (×4): img_044



Urban100 (×4): img_073



Manga109 (×4): PrayerHaNemurenai

Conclusions

- Hybrid Attention Transformer (HAT)를 제안
  - Channel attention 과 self-attention을 결합
    - 더 많은 pixel을 활성화 하여 높은 성능의 고해상도 image 복원
      - Swin Transformer 보다 더 많은 pixel을 활성화
  - Cross window self-attention 제안
    - Cross widow 간의 상호작용을 향상시키기 위함
      - Swin Transformer의 blocking artifact 완화
  - Same-task pre-training
    - 대용량 ImageNet dataset을 사용하여 × 4 SR 모델을 pre-training
    - 이후에 task에 맞는 specific dataset을 fine-tunning

- 정성적, 정량적으로 기존의 SOTA model의 성능을 크게 능가함

감사합니다.