

# Simple Baselines for Image Restoration , Nonlinear activation free network.

## 비선형 함수 필요없는 SOTA 네트워크.

- Abs

최근 이미지 복원 분야에서 상당한 발전이 있었지만 최첨단(SOTA) 방법의 시스템 복잡성도 증가하고 있어 방법을 편리하게 분석하고 비교하는 데 방해가 될 수 있습니다.

본 논문에서는 SOTA 방법을 능가하고 계산적으로 효율적인 간단한 기준선을 제안합니다. 기준선을 더욱 단순화하기 위해 Sigmoid, ReLU, GELU, Softmax와 같은 비선형 활성화 함수는 필요하지 않으며, 곱셈으로 대체하거나 제거할 수 있음을 보여줍니다.

따라서 기준선에서 비선형 활성화 자유 네트워크, 즉 NAFNet을 도출합니다.

- Fig

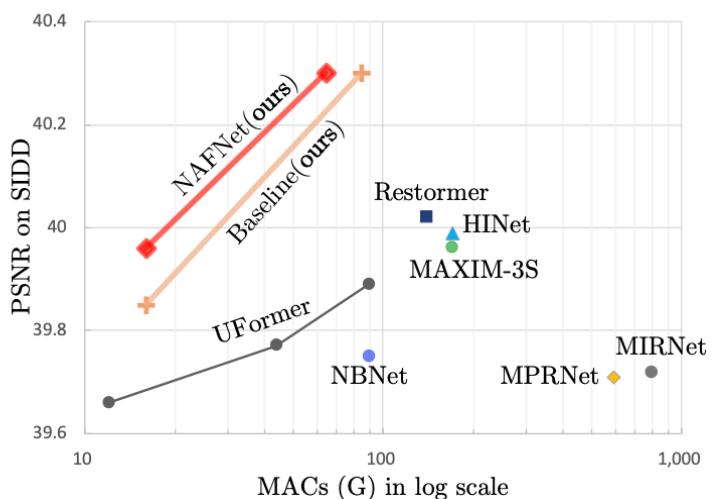
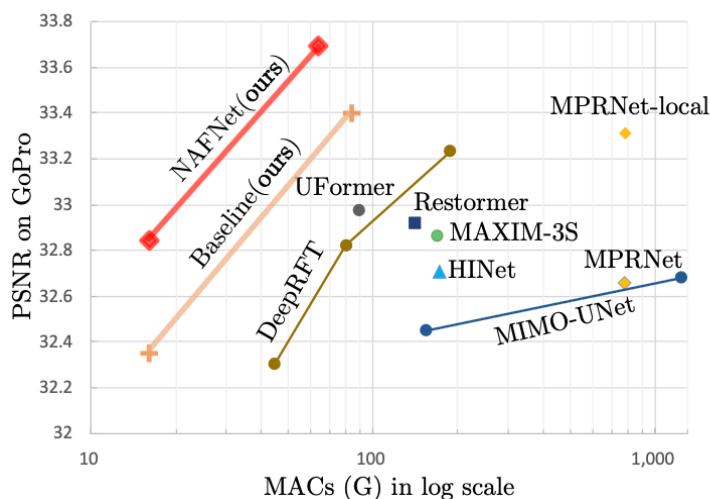
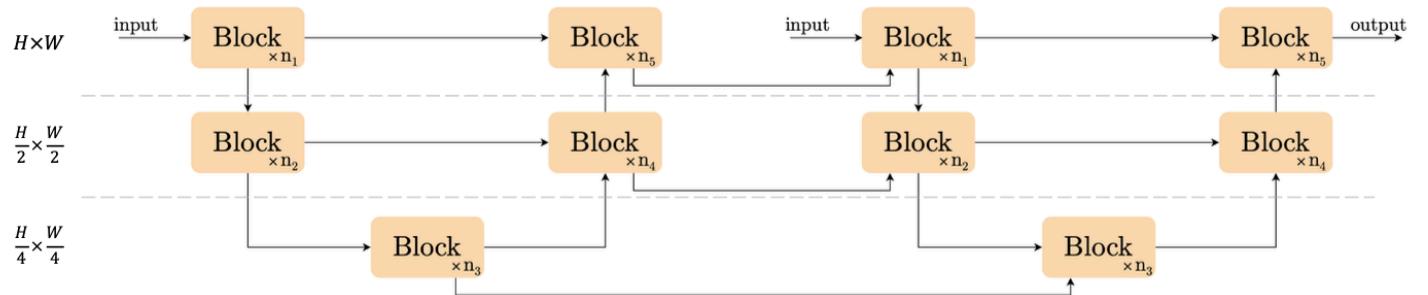


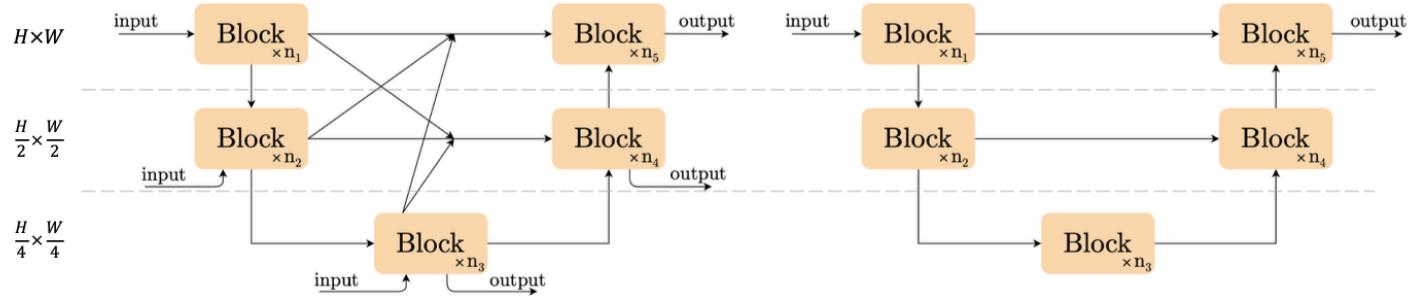
그림 1: 이미지 블러링 제거(왼쪽) 및 이미지 노이즈 제거(오른쪽) 작업에 대한 PSNR vs Computational cost

PSNR : 신호 대 잡음 비.

1. 사용 목적: 생성 혹은 압축된 영상의 화질에 대한 손실 정보를 평가.
2. 의미: 손실이 적을 수록 (화질이 좋을 수록) 높은 값.
3. 주의: 무손실 영상의 경우, MSE가 0이 되기 때문에 PSNR을 정의할 수 없다.



(a) Multi-Stage Architecture



(b) Multi-Scale Fusion Architecture

(c) UNet Architecture(ours)

## 그림 2: 이미지 복원 모델 구조 비교.

(a) MSA[5,37]는 UNet 아키텍처를 직렬로 스택 연결합니다.

(b) MSFA[25,7]는 다양한 스케일로 특징을 융합합니다.

(c) 일부 SOTA 방법에 채택된 UNet Architecture[39,36]. 우리는 이를 아키텍처로 사용합니다.

다음샘플/업샘플된 레이어, Future Fusion module, input/output shortcut 등과 같은 일부 세부 사항은 단순화를 위해 의도적으로 생략되었습니다.

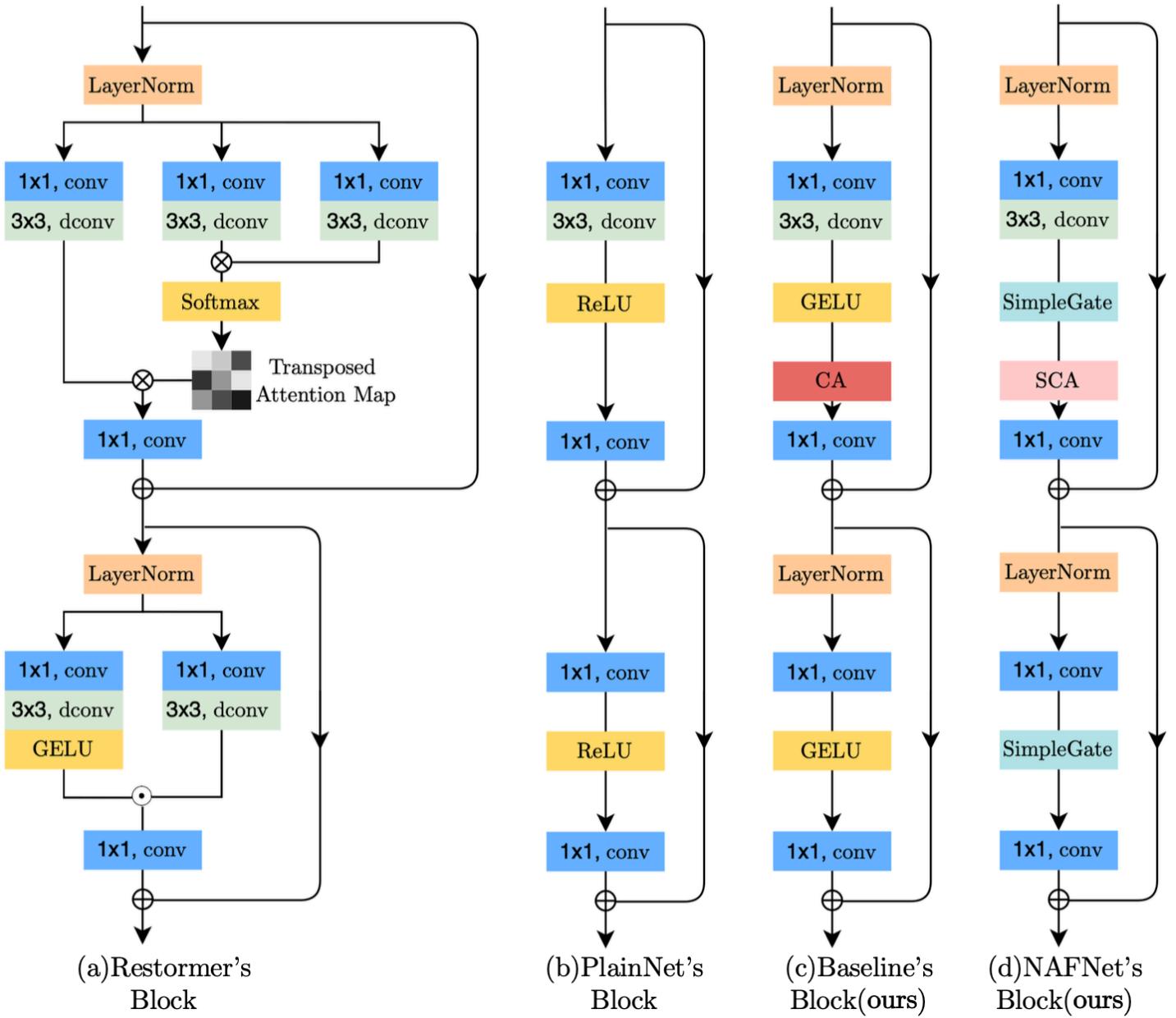


그림 3: 블록 내 구조 비교.

**matrix multiplication, element- wise multiplication/addition**

**dconv : Depthwise Convolution.**

**Nonlinear activation functions :** 노란색 상자로 표시.

**(a) Restormer's block[39],** 단순화를 위해 일부 세부 정보는 생략됩니다.

**(b)** 가장 일반적인 구성 요소를 포함하는 PlainNet's block.

**(c)** 저희가 제안한 기준선입니다. Baseline's block

**(b)와 비교하여 Channel Attention(CA) 및 LayerNorm이 채택되었습니다.**

**측면에서 ReLU는 GELU로 대체됩니다.**

**(d) Our proposed Nonlinear Activation Free Network's block.**

**CA/GELU를 각각 SCA(Simplified Channel Attention) 및 SimpleGate로 대체합니다.**

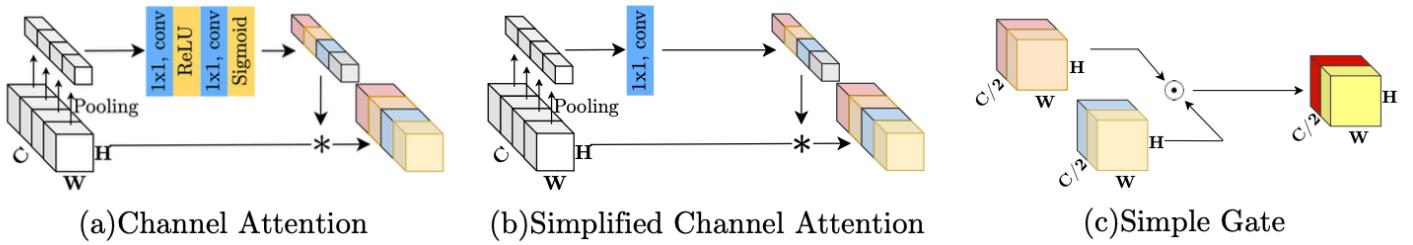


Fig4. (a) Channel Attention[16] (CA)

(b) Simplified Channel Attention (SCA)

(c) Simple Gate (SG).  $\odot/*$ : element-wise/channel-wise multiplication

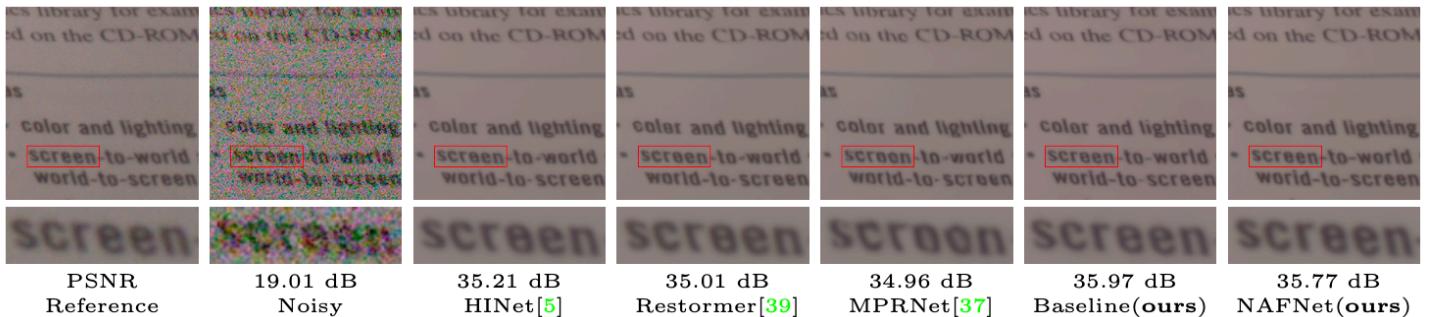


그림 5: SIDD에서 영상노이즈 방법의 정성적 비교[1]

Table 1: Build a simple baseline from PlainNet. The effectiveness of Layer Normalization (LN), GELU, and Channel Attention (CA) have been verified. \* indicates that the training is unstable due to the large learning rate (lr)

	lr	LN	ReLU→GELU	CA	SIDD PSNR SSIM	GoPro PSNR SSIM
PlainNet	$1e^{-4}$				39.29 0.956	28.51 0.907
PlainNet*	$1e^{-3}$				- -	- -
	$1e^{-3}$	✓			39.73 0.959	31.90 0.952
	$1e^{-3}$	✓	✓		39.71 0.958	32.11 0.954
Baseline	$1e^{-3}$	✓	✓	✓	39.85 0.959	32.35 0.956

Table 2: NAFNet is derived from the simplification of baseline, i.e. replacing GELU to SimpleGate (SG), and replacing Channel Attention (CA) to Simplified Channel Attention (SCA).

	GELU→SG	CA→SCA	SIDD PSNR SSIM	GoPro PSNR SSIM	speedup
Baseline	✓		39.85 0.959	32.35 0.956	1.00×
			39.93 0.960	32.76 0.960	0.98×
NAFNet	✓	✓	39.95 0.960	32.54 0.958	1.11×
		✓	39.96 0.960	32.85 0.960	1.09×

표 1: PlainNet에서 간단한 Baseline을 구축합니다.

LN(Layer Normalization), GELU 및 CA(Channel Attention)의 효과가 검증되었습니다.

\* 는 학습률( $\text{lr}$ )이 높아 훈련이 불안정함을 나타냅니다.

표 2: NAFNet은 simplification of baseline, 즉 GELU를 SG(Simple Gate)로 대체하고 CA(Channel Attention)를 SCA(Simplified Channel Attention)로 대체하여 파생됩니다.

Table 3: The effect of the number of blocks. The width is adjusted to keep the computational budget hold. Latency-256 and Latency-720 is based on the input size  $256 \times 256$  and  $720 \times 1280$  respectively, in milliseconds

	# of blocks	SIDD		GoPro		Latency-256	Latency-720
		PSNR	SSIM	PSNR	SSIM		
NAFNet	9	39.78	0.959	31.79	0.951	11.8	154.7
	18	39.90	0.960	32.64	0.951	19.9	151.7
	36	39.96	0.960	32.85	0.959	39.1	177.1
	72	39.95	0.960	32.88	0.961	73.8	230.1

Table 4: Effectiveness of TLC[8] Table 5: Variants of  $\sigma$  in  $\text{SimpleGate}(\mathbf{X}, \mathbf{Y}) = \mathbf{X} \odot \sigma(\mathbf{Y})$  on GoPro[26]

	patches?	TLC?	PSNR	SSIM
NAFNet	✓		33.08	0.963
			33.65	0.966
		✓	33.69	0.967

$\sigma$	SIDD		GoPro	
	PSNR	SSIM	PSNR	SSIM
Identity(ours)	39.96	0.960	32.85	0.960
ReLU	39.98	0.960	32.59	0.958
GELU	39.97	0.960	32.72	0.959
Sigmoid	39.99	0.960	32.50	0.958
SiLU	39.96	0.960	32.74	0.960

표 3: 블록 수에 따른 효과. computational budget을 유지하기 위해 너비가 조정됩니다.

Latency-256 and Latency-720 는 input image size  $256 \times 256$  and  $720 \times 1280$ 을 기반으로 합니다.

표 4 : TLC(Test-time Local Converter) 효율성

표 5 : SimpleGate 변형



Fig. 6: Qualitative comparison of image deblurring methods on GoPro[26]

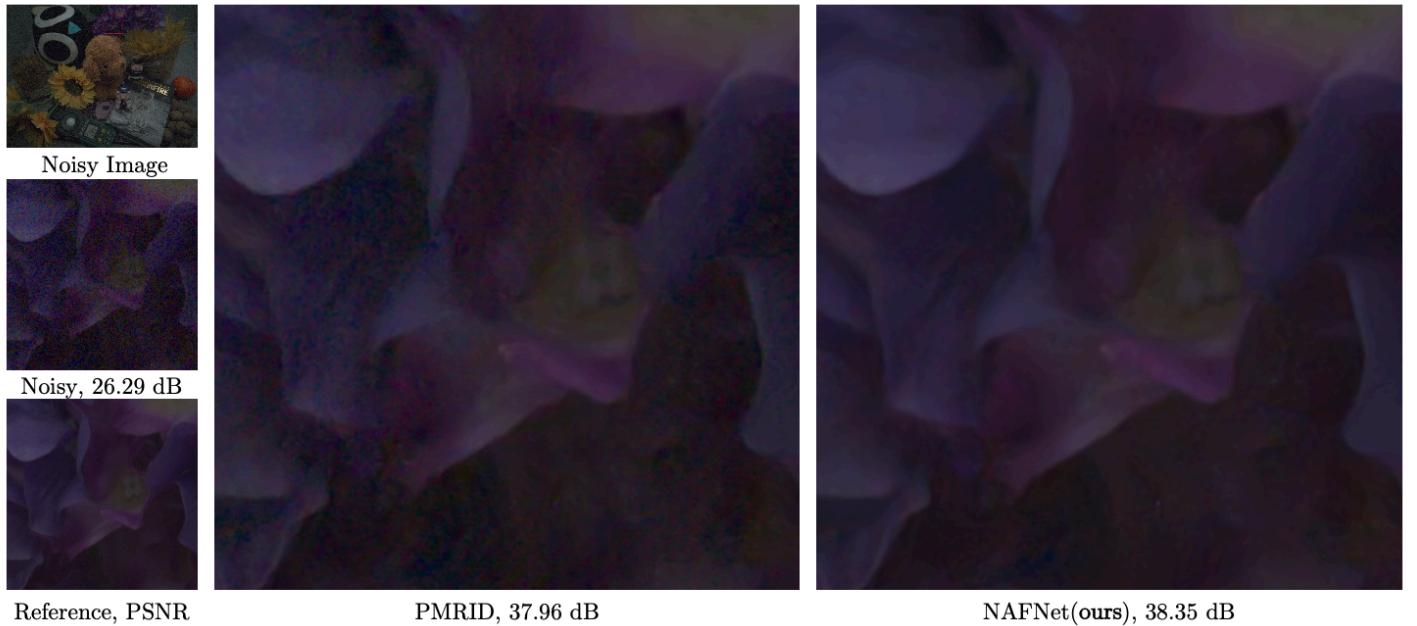
Table 6: Image Denoising Results on SIDD[1]

Method	MPRNet [37]	MIRNet [40]	NBNet [6]	UFormer [36]	MAXIM [32]	HINet [5]	Restormer [39]	Baseline ours	NAFNet ours
PSNR	39.71	39.72	39.75	39.89	39.96	39.99	40.02	40.30	40.30
SSIM	0.958	0.959	0.959	0.960	0.960	0.958	0.960	0.962	0.962
MACs(G)	588	786	88.8	89.5	169.5	170.7	140	84	65

Table 7: Image Deblurring Results on GoPro[26]

Method	MIMO-UNet [7]	HINet [5]	MAXIM [32]	Restormer [39]	UFormer [36]	DeepRFT [25]	MPRNet-local [8]	Baseline ours	NAFNet ours
PSNR	32.68	32.71	32.86	32.92	32.97	33.23	33.31	33.40	33.69
SSIM	0.959	0.959	0.961	0.961	0.967	0.963	0.964	0.965	0.967
MACs(G)	1235	170.7	169.5	140	89.5	187	778.2	84	65

그림 6 : 이미지 디블러링 방법에서 정성적 비교, 표6 : 이미지 디노이징 결과, 표 7 : 이미지 디블러링 결과



**그림 7: PMRID(Practical deep raw image denoising)[35]와 당사가 제안한 NAFNet의 디노이징 효과를 정성적으로 비교합니다.**

- **Intro**

[39]와 [8]은 각각 이미지 노이즈 제거/디블러링에 대해 SIDD[1]/GoPro[26]에서 40.02/33.31dB의 PSNR을 달성합니다.

이러한 방법은 우수한 성능에도 불구하고 높은 시스템 복잡성으로 인해 어려움을 겪습니다. 명확한 논의를 위해 시스템 복잡성을 블록 간 복잡성과 블록 내 복잡성의 두 부분으로 분해합니다.

첫째, 그림 2와 같이 inter-block complexity입니다.

[7,25]는 connections between various-sized feature maps을 소개합니다.

[5,37]은 multi-stage networks이며 후자의 단계는 이전 단계의 결과를 개선합니다.

둘째, intra-block complexity, 즉 블록 내부의 다양한 설계 선택입니다.

Multi-Dconv Head Transposed Attention Module and Gated Dconv Feed-Forward Network in [39] (as we shown in Figure 3a), Swin Transformer Block in [22], HINBlock in [5] 등

위의 사실을 바탕으로 network with low inter-block and low intra-block complexity can achieve SOTA performance?라는 자연스러운 질문이 제기됩니다.

첫 번째 조건(low inter-block complexity)을 달성하기 위해 본 논문에서는 single-stage UNet as architecture로 채택하고(일부 SOTA 방법[39,36] 두번째 조건을 따릅니다.

이를 위해 start with a plain block with the most common components, 즉 컨볼루션, ReLU 및 shortcut[14]로 시작합니다.

Plain block에서 SOTA 방법의 구성 요소를 추가/교체하고 이러한 구성 요소가 얼마나 많은 성능을 얻는지 확인합니다.

광범위한 절제 연구를 통해 그림 3c와 같이 SOTA 방법을 능가하고 계산적으로 효율적인 간단한 기준선을 제안합니다. GELU[15] 및 Channel Attention Module[16](CA)을 포함하여 baseline은 더욱 단순화될 수 있습니다.

저희는 baseline의 GELU가 Gated Linear Unit[10](GLU)의 특수한 경우로 간주될 수 있음을 밝히고, 이를 통해 simple gate, 즉 element-wise product of feature maps으로 대체할 수 있음을 경험적으로 입증합니다.

또한 CA와 형태의 GLU의 유사성을 드러내며, CA의 nonlinear activation functions도 제거할 수 있습니다. 결론적으로, 간단한 기준선은 NAFNet으로 알려진 nonlinear activation free network로 더욱 단순화될 수 있습니다.

저희는 주로 [5,39,37]에 이어 이미지 노이즈 제거를 위한 SIDD[1], 이미지 디블러링을 위한 GoPro[26]에 대한 실험을 수행합니다.

주요 결과는 그림 1에 나와 있으며, 제안된 기준선과 NAFNet은 계산 효율성과 함께 SOTA 결과를 달성합니다.

이 논문의 기여 내용은 다음과 같이 요약됩니다:

1. SOTA 방법을 분해하고 필수 구성 요소를 추출하여 시스템 복잡성이 낮은 Baseline(그림 3c)을 형성하는데, 이는 그림 1과 같이 이전 SOTA 방법을 초과할 수 있고 Computational cost이 더 낮습니다. 이는 연구자들이 새로운 아이디어에 영감을 주고 편리하게 평가할 수 있도록 촉진 할 수 있습니다.
2. GELU, Channel Attention to Gated Linear Unit 간의 연결을 밝힘으로써 Nonlinear activation functions(예: Sigmoid, ReLU, GELU)를 제거하거나 교체하여 Baseline을 더욱 단순화하고 nonlinear activation free network, 즉 NAFNet을 제안합니다. 단순화되지만 Baseline과 일치하거나 능가할 수 있습니다. 저희가 아는 한, 이 작업은 SOTA 컴퓨터 비전 방법에 Nonlinear activation functions가 필요하지 않을 수 있음을 보여주는 첫 번째 작업입니다. 이 작업은 SOTA 컴퓨터 비전 방법의 설계 공간을 확장할 수 있는 잠재력을 가질 수 있습니다.

- Conclu

SOTA 방법을 분해하여 필수 구성 요소를 추출하고 naive PlainNet(Resnet에 층만 다른 네트워크)에 채택 합니다.

얕은 Baseline은 이미지 노이즈 제거 및 이미지 디블러링 작업에서 SOTA 성능에 도달합니다.

Baseline을 분석하면 Nonlinear activation functions를 완전히 대체하거나 제거할 수 있다는 점을 알 수 있습니다.

이를 통해 nonlinear activation free network인 NAFNet을 제안합니다.

단순화되었지만 성능은 Baseline과 같거나 더 우수합니다.

저희가 제안한 Baseline은 연구자들이 아이디어를 평가하는 데 용이할 수 있습니다.

또한, 이 작업은 nonlinear activation functions가 SOTA 성능을 달성하는 데 필요하지 않다는 것을 보여주기 때문에 미래의 컴퓨터 비전 모델 설계에 영향을 미칠 가능성이 있습니다.