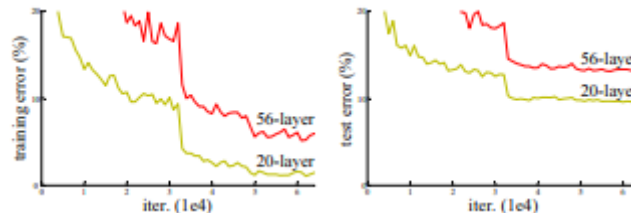


Deep Residual Learning for Image Recognition

잔차를 이용한 잔차학습을 이용해 깊은 신경망에서도 training이 쉽게 이뤄질 수 있다는 것을 보임

Degradation Problem - network가 깊어질수록 accuracy가 떨어지는 문제(overfitting 문제 아님)



모든 시스템이 optimize하기 쉬운 것이 아니라는 뜻 => 얇은 구조와 더 깊은 아키텍처를 비교

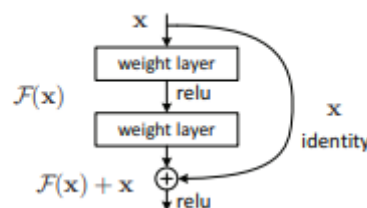
Deep residual learning framework 개념 도입

얇은 레이어가 그 다음 레이어에 바로 적합되는 것이 아닌 잔차의 mapping에 적합하도록 함.

비선형적 layer $F(x) = H(x) - x \Rightarrow H(x) = F(x) + x \Rightarrow$ Shortcut Connection과 동일 \Rightarrow 하나 또는 하나 이상의 레이어를 skip하게 만들

residual mapping이 기존의 mapping보다 optimize하기 쉽다고 가정

Identify Short Connection - 추가적인 파라미터 필요x. 복잡한 곱셈 연산 필요x



input x 가 model $F(x)$ 를 거치며 identity인 x 가 더해져 output으로 $F(x) + x$ 나옴

목표 : 쉬운 optimization, accuracy 높이기

$H(x)$ 를 기본 mapping으로 간주하고 x 가 input일 때 다수의 비선형 레이어가 복잡한 함수를 점근적으로 근사할 수 있다고 가정하면 $H(x) - x$ 를 무의식적으로 근사할 수 있음

$$y = F(x, \{W_i\}) + x.$$

$$y = F(x, \{W_i\}) + W_s x.$$

단순 덧셈으로 인해 복잡한 구조와 연산이 필요없다는 것이 핵심. 첫 번째 식의 경우 $F = W_2 \sigma(W_1 x)$ 를 간소화한 모양. x 는 relu함수 σ 를 한 번 통과했고 bias 생략해서 나타냄. W_1 , W_2 등의 linear projection을 사용해 같은 차원으로 만들어줌



Plain Network : VGG Net 참고. 동일한 feature map 사이즈를 갖게 하기 위해 layer는 동일한 수의 filter를 가지게 함. feature map 사이즈가 절반이 되면 filter 수 2배가 됨. Conv layer는 3*3의 filter, stride 2로 downsampling, global average pooling layer 사용. 마지막에는 softmax로 1000-way-fully-connected layer 통과.

-> 18-layer보다 34-layer 성능이 더 낮음

-> BN으로 학습되었으므로 non-zero variance를 가지고 optimization이 어려운 이유가 vanishing gradient 때문이 아닌 것을 알 수 있음

-> 기하급수적으로 수렴이 어려워지는 것을 training error의 reducing에서 어려운 점으로 봄.

Residual Network : Plain Network를 기본 바탕으로 하나 shortcut connection 개념 도입.

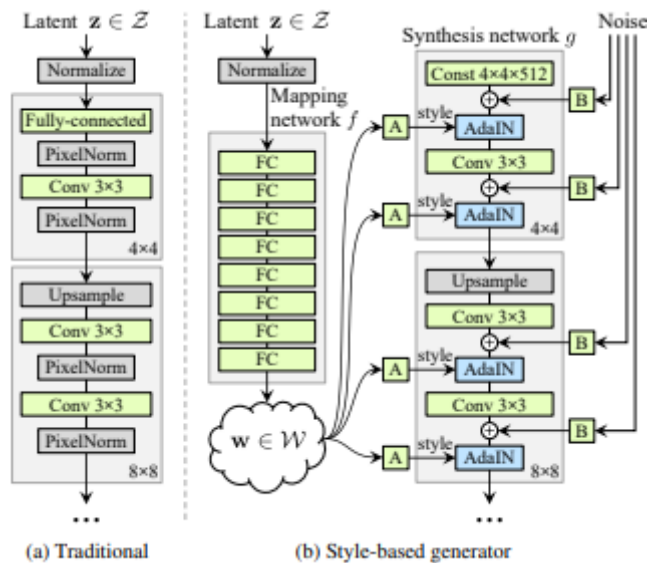
Identity Shortcut은 input과 output을 같은 차원으로 맞춰야 함. 차원을 늘리기 위해 0을 넣어서 padding

-> 18-layer보다 34-layer ResNet이 성능이 좋음. 더 낮은 training error.

-> 18-layer끼리 비교했을 때 accuracy는 비슷하나 18-layer ResNet의 수렴이 더 빠른 걸로 보아 ResNet이 SGD를 사용한 optimization이 더 쉽다는 것을 알 수 있음.

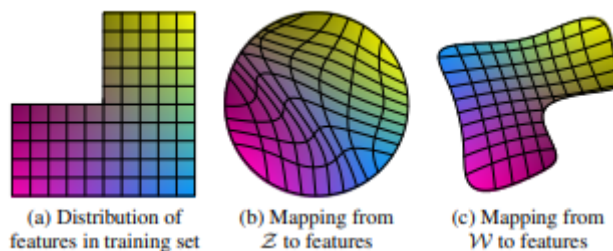
	plain	ResNet
18 layers	27.94	27.88
34 layers	28.54	25.03

A Style-Based Generator Architecture for Generative Adversarial Networks



traditional GAN의 경우 latent z 를 바로 input으로 넣어준 반면, StyleGAN에서는 학습된 constant 값을 넣어줌. 또한 새롭게 mapping network와 noise가 추가됨.

mapping network를 통해 w 를 매핑한 후, w 를 넣은 이유 => disentanglement



disentanglement : 각 변수가 하나의 변동 요인을 제어하는 linear subspaces로 구성된 latent space

\mathcal{W} 를 feature에 매핑하는 경우 \mathcal{W} 는 \mathcal{Z} 처럼 고정된 분포를 따르지 않음. sampling density는 학습된 piecewise continuous mapping $f(z)$ 에 정해짐. 따라서 warping이 많이 일어나지 않으므로 factors of variation은 더욱 linear하고, disentangled함.

z 를 중간 latent space \mathcal{W} 에 매핑을 한 뒤 w 는 학습된 affine transform A 를 거쳐 $y=(y_s, y_b)$ 로 변형됨. 이 style들은 AdaIN operation을 control

$$\text{AdaIN}(\mathbf{x}_i, \mathbf{y}) = \mathbf{y}_{s,i} \frac{\mathbf{x}_i - \mu(\mathbf{x}_i)}{\sigma(\mathbf{x}_i)} + \mathbf{y}_{b,i},$$

noise를 통해 결정하는 것 : stochastic variation

stochastic variation : perception of the image에 영향을 주지 않고 randomized될 수 있는 부분

disentanglement를 측정하는 metric : Perceptual path length, Linear separability

Method	Path length		Separa- bility
	full	end	
B Traditional generator Z	412.0	415.3	10.78
D Style-based generator W	446.2	376.6	3.61
E + Add noise inputs W	200.5	160.6	3.54
+ Mixing 50% W	231.5	182.1	3.51
F + Mixing 90% W	234.0	195.9	3.79

Perceptual path length - latent space에서 interpolation 했을 때, 얼마나 큰 변화가 있는지 측정

Linear separability - latent space 점들이 linear hyperplane으로 잘 구분되는지 측정

Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising

일반적으로 specific task를 위해 Deep CNN model을 학습시킬 때 2가지 단계를 가짐.

1. 네트워크 구조 디자인
2. training data로부터 학습하는 model

네트워크 구조 디자인을 위해 VGG network를 image denoising에 맞게 변형, sota denoising method에서 사용한 효과적인 patch sizes를 기반으로 network의 depth를 설정. model 학습을 위해 residual learning formulation 사용. 빠른 학습과 향상된 denoising performance를 위해 batch normalization 포함.

DnCNN Network Depth

- Convolution filter size : 3x3, pooling layer 제거
- Denoising neural networks의 receptive field size는 effective patch size와 관련있음. high noise level은 더 많은 context information을 잡아내기 위해 큰 effective patch size 요구.

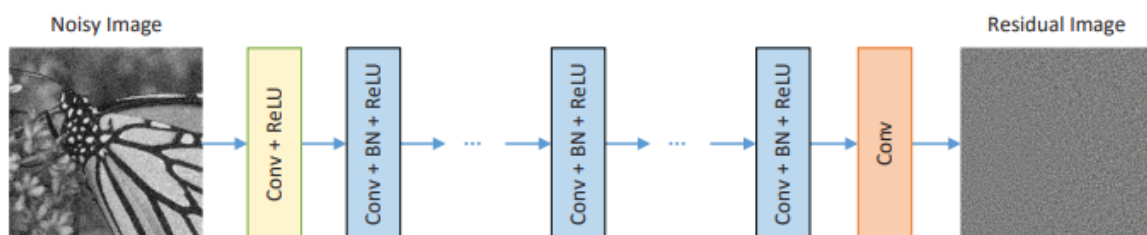
TABLE I
THE EFFECTIVE PATCH SIZES OF DIFFERENT METHODS WITH NOISE LEVEL $\sigma = 25$.

Methods	BM3D [2]	WNNM [13]	EPLL [33]	MLP [24]	CSF [14]	TNRD [16]
Effective Patch Size	49×49	361×361	36×36	47×47	61×61	61×61

DnCNN Architecture

CNN 사용. 우선 Ground truth image(Y) 구함. AWGN noise를 더하여 noisy Image X 만듦. noisy Image X에 CNN 네트워크를 이용하여 passed 이미지를 만들고, 원래 이미지 Y에서 결과 이미지를 감산 연산하여 residual image 만듦. 최종적으로 원래 이미지와 residual image의 MSE를 계산하여 CNN 네트워크의 Optimization에 반영. zero padding하는 것으로 기존의 CSF나 TNRD의 가장자리 Artifact문제 해결.

목표 : MLP method나 CSF method와 같은 Discriminative denoising models은 latent clean image를 예측하기 위해 mapping function을 학습시키는 것

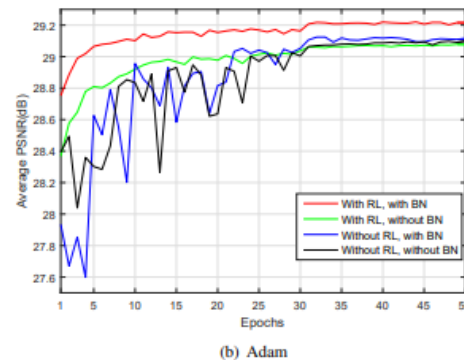
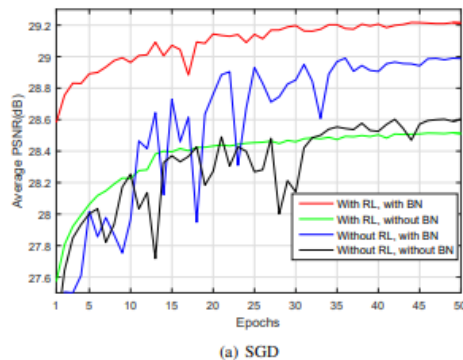


특징 : $R(y)$ 를 학습시키기 위해 Residual learning formulation 사용.

학습 속도와 denoising performance를 향상시키기 위해 batch normalization 포함.

많은 low level vision applications은 보통 output image size를 input image size와 같게 유지할 것을 요구하는데, 이로 인해 Boundary artifacts가 생겨날 수 있음.

DnCNN : middle layers의 각 feature map들이 계속해서 input image와 같은 사이즈를 가질 수 있도록 convolution 전에 바로 zero padding을 해주었음. x 를 예측하기 위해 original mapping $F(y)$ 를 학습시키거나, v 를 예측하기 위해 residual mapping $R(y)$ 를 학습시키는 데에 사용될 수 있음. Original mapping이 identity mapping에 더 가까울 때 residual mapping은 더욱 최적화되기 쉽고, noisy observation y 는 residual image v 보다 latent clean image x 에 더 가까움. 따라서 $R(y)$ 보다 $F(y)$ 가 identity mapping에 더 가깝고, residual learning formulation은 image denoising에 더욱 적합.



- residual learning formulation을 사용할 경우, original mapping learning보다 더 빠르고 안정적으로 수렴됨.
- residual learning formulation과 batch normalization을 함께 사용할 경우, original mapping보다 빠르게 수렴하고 더 좋은 denoising performance를 나타내는데, 특히 SGD와 Adam optimization algorithms이 이 network가 best 결과를 가질 수 있도록 도움.