

## <DCGAN 논문 리뷰>

### - Abstract

- CNN을 이용한 지도학습은 컴퓨터비전 분야에서 많이 채택이 되어왔지만, 비지도학습은 덜 주목받아 왔음. DCGAN을 통해 그 간극을 줄이고자 함.

### - Introduction

- 논문을 통해 기여할 점

- 1) 대부분의 상황에서도 안정적으로 학습이 되는 Convolution 구조 제안
- 2) 다른 비지도 학습 방법들과 비교하기 위해서, 이미지 분류 과제에 학습되었던 discriminator 사용
- 3) GAN에 의해 학습된 필터 시각화 및 특정 물체를 그리도록 학습된 특정 필터를 보여줌.
- 4) generator가 벡터 연산적인 특성으로 쉽게 조작하여 샘플을 생성하는 것을 보여줌.

### - Related Work

- 기존 GAN의 한계

- 1) 모델의 성능을 평가할 기준이 모호하기에 clustering 등의 방식을 모델의 성능을 평가하고자 했음.
- 2) 생성 데이터의 해상도: 자연스럽게 생성하는 것이 어려웠음.
- 3) Black-box method: Neural net 자체의 한계. 그러나 deconvolution을 사용하고 maximal activation을 필터링함으로써 각 convolution의 필터의 목적을 대략적으로 알 수 있음.

### - Approach and Model Architecture

#### Architecture guidelines for stable Deep Convolutional GANs

- Replace any pooling layers with strided convolutions (discriminator) and fractional-strided convolutions (generator).
- Use batchnorm in both the generator and the discriminator.
- Remove fully connected hidden layers for deeper architectures.
- Use ReLU activation in generator for all layers except for the output, which uses Tanh.
- Use LeakyReLU activation in the discriminator for all layers.

- fractional-strided convolution: input에 padding을 하고 convolution을 하면서 오히려 크기가 커지는 특징이 있음.
- batch normalization 적용, 그러나 generator의 output layer와 discriminator의 input layer에는 BN 적용 X.

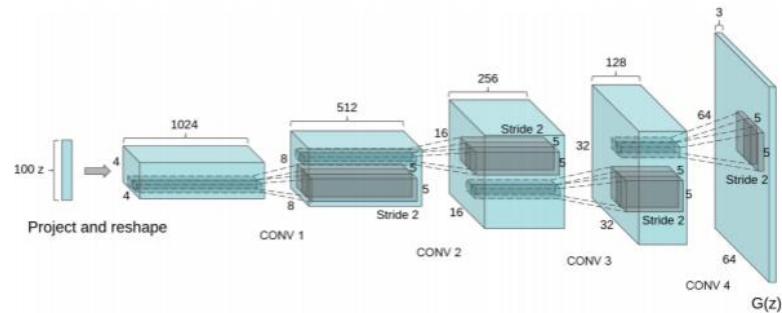


Figure 1: DCGAN generator used for LSUN scene modeling. A 100 dimensional uniform distribution  $Z$  is projected to a small spatial extent convolutional representation with many feature maps. A series of four fractionally-strided convolutions (in some recent papers, these are wrongly called deconvolutions) then convert this high level representation into a  $64 \times 64$  pixel image. Notably, no fully connected or pooling layers are used.

#### <generator의 구조>

#### - DCGAN 결과

- generator가 이미지를 외워서 보여주는 것이 아님을 보여줌.
  - image de-duplication: autoencoder를 통해 code로 변환하고 이 code와 가까운 값들을 제거하여 precision을 높이고 false positive 비율을 0.01 이하로 하는 결과를 얻음.



Figure 2: Generated bedrooms after one training pass through the dataset. Theoretically, the model could learn to memorize training examples, but this is experimentally unlikely as we train with a small learning rate and minibatch SGD. We are aware of no prior empirical evidence demonstrating memorization with SGD and a small learning rate.

- walking in the latent space: generator의 input 공간인 latent space에서 이동할 때 부드럽게 변화해야 함. 급작스러운 변화가 일어날 경우 memorization으로 판단함.



(벽이었던 곳이 창문으로 바뀌어가는 과정)

- black box 일정부분 해소: discriminator가 학습한 feature의 어떤 부분을 input의 어떤 부분을 활성화하여 인식하고 있는지를 보여줌.

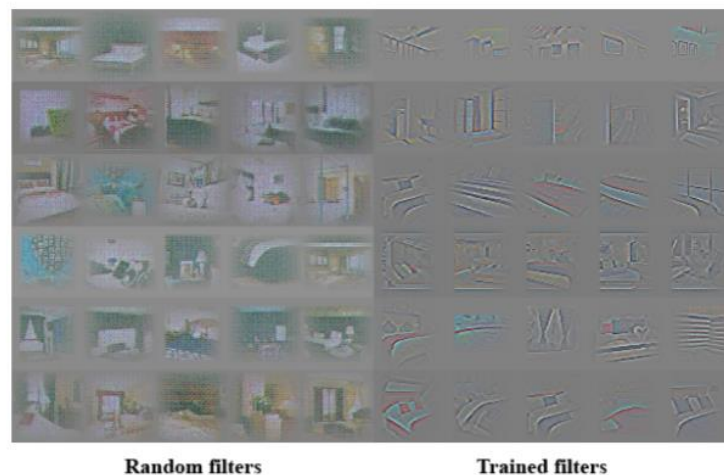


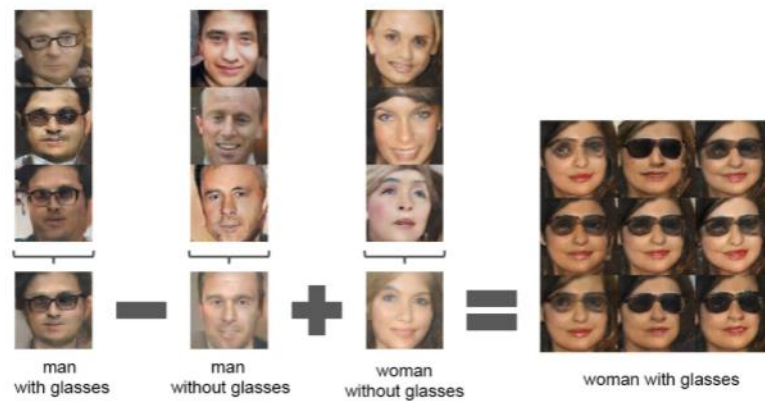
Figure 5: On the right, guided backpropagation visualizations of maximal axis-aligned responses for the first 6 learned convolutional features from the last convolution layer in the discriminator. Notice a significant minority of features respond to beds - the central object in the LSUN bedrooms dataset. On the left is a random filter baseline. Comparing to the previous responses there is little to no discrimination and random structure.

- 학습한 특징 잊어버리기 가능: 각 특징을 맡고 있는 filter를 drop시켜서 이미지를 생성하면 실제 이미지에서 해당 특징이 사라짐.



Figure 6: Top row: un-modified samples from model. Bottom row: the same samples generated with dropping out "window" filters. Some windows are removed, others are transformed into objects with similar visual appearance such as doors and mirrors. Although visual quality decreased, overall scene composition stayed similar, suggesting the generator has done a good job disentangling scene representation from object representation. Extended experiments could be done to remove other objects from the image and modify the objects the generator draws.

- word2vec과 같이 이미지도 벡터의 연산이 가능



- scale, rotation, position과 같은 특성 이해한다고 볼 수 있음.



Figure 8: A "turn" vector was created from four averaged samples of faces looking left vs looking right. By adding interpolations along this axis to random samples we were able to reliably transform their pose.

## - Conclusion

- GAN의 불안정함을 해소하긴 했지만, 오래 학습시킬 경우 oscillating mode 등의 불안정한 모습이 남아있음.