StyleGAN

A Style-Based Generator Architecture for Generative Adversarial Networks

Abstract

- interpolation quality와 disentanglement를 정량화 하기위해 어떤 generator architecture에도 적용가능한 2개의 자동화된 방법을 제안했다.
- 새로운 아키텍처는 자동으로 학습되고 높은 수준의 속성(포즈와 인간의 얼굴을 학습시켰을 때 identity)과 생성된 이미지(주근깨, 머리카락 등)의 stochastic variation을 비지도 분리하게 한다.
- 또한 합성이 intuitive, scale-specific control하게 되도록 한다.

Introduction

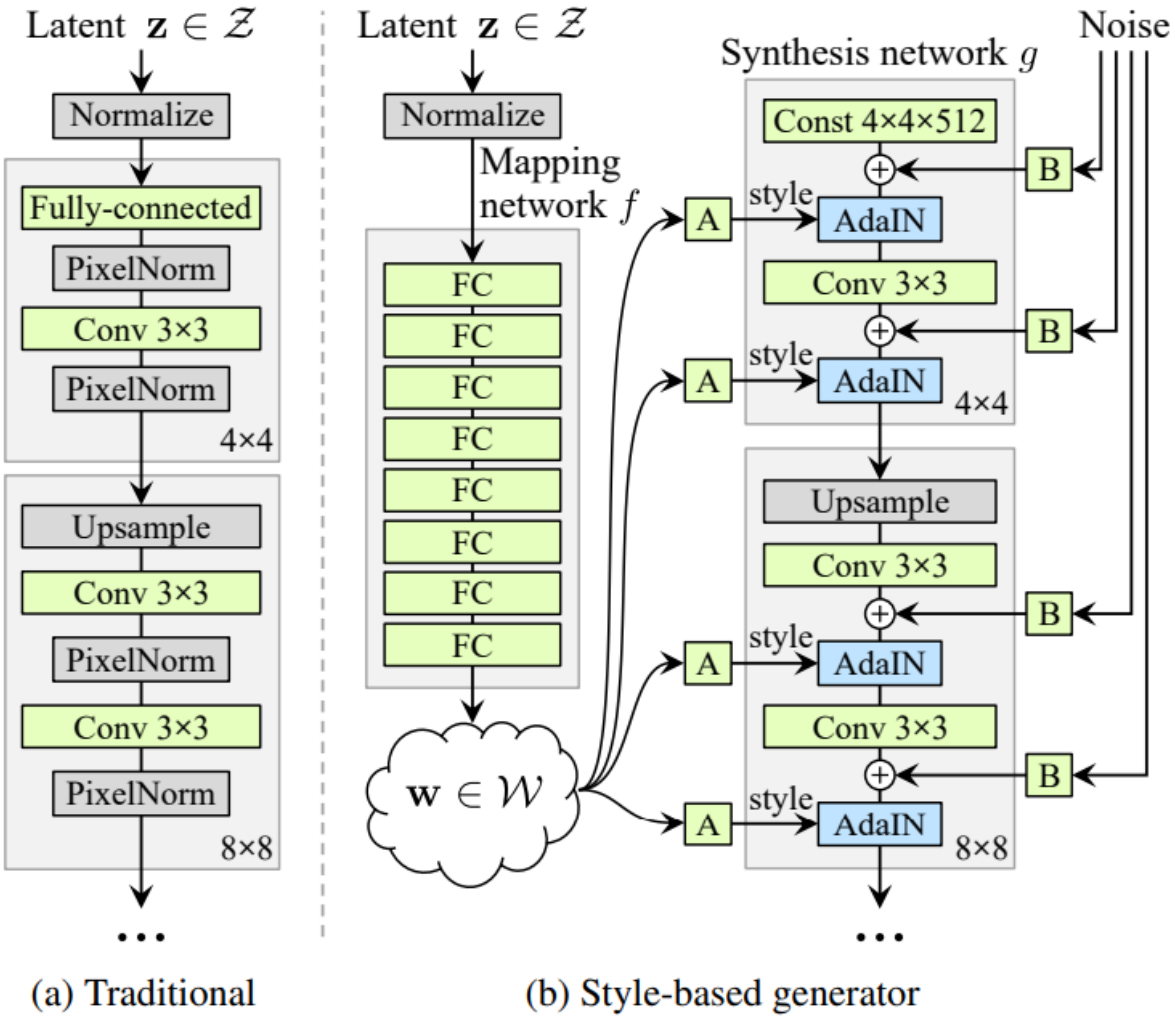
최근 많은 노력에도 불구하고, 이미지 합성의 다양한 측면(origin of stochastic features 등)의 이해는 여전히 부족하다. latent space의 속성 또한 잘 이해되지 못하였고, 일반적으로 입증된 latent space의 보간은 여러 generator을 비교하는 정량적 방법을 제공하지 않는다.

Generator은 학습된 constant input에서 시작해 이미지의 스타일을 각 latent code에 기초한 convolution layer마다 조정한다. 따라서 다른 scale에 있는 이미지의 특징의 strength를 컨트롤 할 수 있게 한다. 이 아키텍쳐의 변화는 네트워크에 직접 주입된 노이즈와 합쳐져 자동이고 고수준의 속성을 생성된 이미지의 stochastic variation에서 비지도 분리가 되게 하며, 직관적인 scale별 믹싱과 보간작업이 가능하게 한다.

Generator은 input latent code를 중간의 latent space에 포함시킨다(변동 요인이 네트워크에 표현되는 방식에 큰 영향을 미침). 원래라면 input latent space는 학습 데이터에 확률 밀도를 따라야 하지만, 본 논문의 intermediate latent space는 그 제약조건에서 자유롭다(따라서 disentangle 될 수 있음).

latent space의 degree를 측정하는 이전의 방법은 본 논문의 case에 적용되기가 어려워 본 논문의 저자들은 generator의 이러한 측면을 정량화하기 위해 2개의 새로운 자동 metric을 제안한다. 이 metric들(perceptual path length and linear separability)을 통해 기존 generator 아키텍쳐와 비교했을 때 본 논문의 generator은 다양한 변형 요인에 대해 더욱 linear하고 덜 entangle된 표현을 보여준다.

Style-based generator



기존에는 latent code가 input layer을 통해 generator에게 제공되었다. 본 논문에서는 input layer을 모두 생략하고 학습된 constant에서 시작하였다. input latent space Z안에 있는 latent code z가 주어지면 네트워크 $f : Z \rightarrow W$ 는 먼저 $w \in W$ 를 생성한다. 단순화를 위해 두 space의 dimensionality를 512로 지정하고, mapping f 를 8-layer MLP로 구현하였다. 그 다음 학습된 아핀변환은 w를 합성 네트워크 g의 각 convolution layer 후에 adaptive instance normalization를 제어하는 스타일 $y = (y_s, y_b)$ 로 specialize한다.

📘 아핀변환(Affine Transform)

Adaptive instance normalization

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

- Feature map인 x_i 가 별도로 정규화되고, style y 중 대응되는 스칼라 구성요소들을 통해 scaled되고 biased된다. 따라서 y 의 차원은 해당 layer의 feature map 수의 2배이다.

본 논문에서는 example image 대신 벡터 w 에서 공간적으로 변하지 않는 스타일인 y를 계산한다.

결론은 generator에게 명시적인 noise input을 사용하여 확률적인 세부 정보를 생성하는 직접적인 수단을 제공한다는 것이다. 명시적인 noise input는 uncorrelated Gaussian noise를 포함하는 single-channel 이미지이며 합성 네트워크의 각 layer에 dedicated noise 이미지를 공급한다. 또한 이 이미지는 모든 feature map에 “learned perfeature scaling factors”을 통해 broadcast 되고, 대응하는 convolution의 output에 더해진다.

Properties of the style-based generator

본 논문의 generator 아키텍쳐는 이미지 합성이 스타일에 대한 scale-specific 수정을 통해 컨트롤 가능하게 한다. 학습된 분포에서 각 스타일에 대한 샘플을 추출하는 방법으로 mapping network와 아핀변환을 사용할 수 있으며, 스타일 컬랙션을 기반으로 새로운 이미지를 생성하는 방법으로 합성 네트워크(synthesis network)를 사용할 수 있다. 각 스타일의 효과는 네트워크에 localize 되어진다.

Style mixing

본 논문에서는 스타일이 localize할 수 있도록 mixing regularization을 제안하였다(주어진 이미지의 퍼센티지가 2개의 랜덤한 latent code를 사용해 생성됨). 이미지를 생성할 때 합성 네트워크의 랜덤하게 선태고딘 포인트에서 하나의 latent code를 다른 latent code로 변환시킨다(style mixing). 즉, 2개의 latent code인 z_1, z_2 를 mapping network를 통해 실행하고 corresponding한 w_1, w_2 가 각각 crossover point 앞, 뒤에서 적용된다. 이를 통해 네트워크기 인접한 스타일이 관계있다고 가정하지 않도록 해준다.

Stochastic variation

기존의 generator이 stochastic variation을 구현하는지 본다면, input layer으로 input값이 유일하게 들어가기에 네트워크는 필요할 때마다 이전 activation에서 공간적으로 변화하는 pseudorandom number을 생성하는 방법을 만들어내야 한다. 다만 이러한 방법은 완벽하지 못하다. 생성된 이미지에서 반복적으로 나타나는 패턴을 통해 이 방법이 완벽하지 못하다

는 것을 알 수 있다. 본 논문의 아키텍쳐는 이러한 문제를 각 convolution 이후에 per-pixel noise를 더함으로서 회피한다.

Separation of global effects from stochasticity

스타일 기반 generator에서 스타일은 모든 feature map이 동일한 값으로 scaled되고 biased되기 때문에 전체 이미지에 영향을 준다. 따라서 global effect인 포즈, 조명, 배경 스타일 등은 일관되게 컨트롤 될 수 있다.

노이즈는 각 픽셀에 독립적으로 더하여지기에 stochastic variation을 컨트롤 하는데 이상적이다. 예로 네트워크가 noise를 통해 포즈를 취하려고 한다면 판별기에 의해 페널티가 부과되는 공간적으로 일치하지 않는 결정으로 이어질 것이다. 따라서 네트워크는 명시적인 지침 없이 global 및 local channel을 적절하게 사용하는 방법을 배운다.

Disentanglement studies

Disentanglement의 목표는 linear subspace로 구성된 공간으로 구성된 latent space이다. 각 subspace는 하나의 변동 요인을 제어한다.

Perceptual path length

latent space vector의 보간은 이미지에 비선형 변화를 가져올 수 있다. 예로 두 끝점에는 없는 기능이 보간 경로의 중간에 나타날 수 있다는 것이다. 이는 latent space가 얽혀있고, 변동 요인이 잘 분할되지 못하였다고 볼 수 있다. 이러한 현상을 정량화하기위해, 본 논문에서는 latent space에서 보간을 진행할 때 이미지가 얼마나 급격한 변화를 겪는지 측정한다. 결국 덜 curved된 latent space는 더욱 curved된 latent space보다 더 부드러운 변환을 가져온다는 것이다.

Metric의 기초로 2개의 VGG16 임베딩 사이의 가중치 차이로 계산되는 지각 기반 pairwise 이미지 거리를 사용한다. 가중치는 metric이 인간의 지각 유사성 판단과 일치하도록 fit된다. 만약 latent space interpolation path를 linear segment로 세분하면 이 segment path의 총 perceptual length는 image distance metric에 의해 report되는 각 segment의 perceptual differences의 합으로 정의할 수 있다.

Perceptual path 길이의 자연적인 정의는 무한한 미세 세분화 하에서 이 합계의 limit이 될 것이지만, 본 논문에서는 epsilon$(\epsilon = 10^{-4})$을 사용해 근사화한다. latent space에서 평균 perceptual path인 Z 는 가능한 모든 끝점에 대해

$$l_{\mathcal{Z}} = \mathbb{E}\left[\frac{1}{\epsilon^2} d\big(G(\text{slerp}(\mathbf{z}_1, \mathbf{z}_2;\, t)),\; G(\text{slerp}(\mathbf{z}_1, \mathbf{z}_2;\, t+\epsilon))\big)\right],$$

이다.

- $z_1, z_2 \sim P(z), t \sim U(0,1), G$ (generator)
- $d(\cdot,\cdot)$: 결과 이미지들의 Perceptual distance를 평가함.
- slerp : Spherical interpolation
- 얼굴의 특징에 집중하기 위해 생성된 이미지들을 얼굴만 포함하도록 자른다.
- d 가 2차이기에 ϵ^2 로 나누어준다.
- 본 논문에서는 10000개의 샘플로 기대값을 계산하였다.

W 안의 평균 perceptual path 길이 계산도 유사한 방식으로 진행된다.

$$l_{\mathcal{W}} = \mathbb{E}\left[\frac{1}{\epsilon^2} d\big(g(\text{lerp}(f(\mathbf{z}_1), f(\mathbf{z}_2);\, t)),\; g(\text{lerp}(f(\mathbf{z}_1), f(\mathbf{z}_2);\, t+\epsilon))\big)\right],$$

유일한 차이는 W 공간에서 보간이 일어난다는 것이다. 왜냐하면 W안의 벡터가 어떤 fashion에서도 정규화되지 않기 때문이다(lerp).

Linear separability

latent space가 충분히 disentangled 되었다면 다양한 factor과 correspond 하는 direction vector을 찾는 것이 가능해야한다. 따라서 본 논문에서는 latent-space point가 linear hyperplane에 따른 2개의 dispinct 세트를 통해 얼마나 잘 나누어졌는지를 측정하는 metric를 제안하였다(각각의 세트는 이미지의 특정한 binary attribute에 correspond된다).

생성된 이미지의 레이블을 붙이기 위해, auxiliary classification network를 학습시킨다.

각각의 attribute에 latent space를 기반으로 SVM을 통해 label을 예측한다.