



# StyleGAN: A Style-Based Generator Architecture for GANs

## Abstract

우리는 스타일을 변환해주는 적대적 생성 신경망을 제안합니다. 이 신경망은 큰 특징과 작은 특징에 대해 학습할 수 있으며 직관적이고 그 정도를 조절할 수 있습니다. 새로운 생성기는 평가지표를 발전시키며 더욱 이미지를 잘 만들고 특징 구분을 잘합니다. 특징 구분에 대해서 더욱 잘 파악하기 위해 생성 모델에 적용 가능한 두 가지 새로운 방식을 제안하며 마지막으로 사람 얼굴에 대한 퀄리티 높은 데이터 셋을 제시합니다.

## Introduction

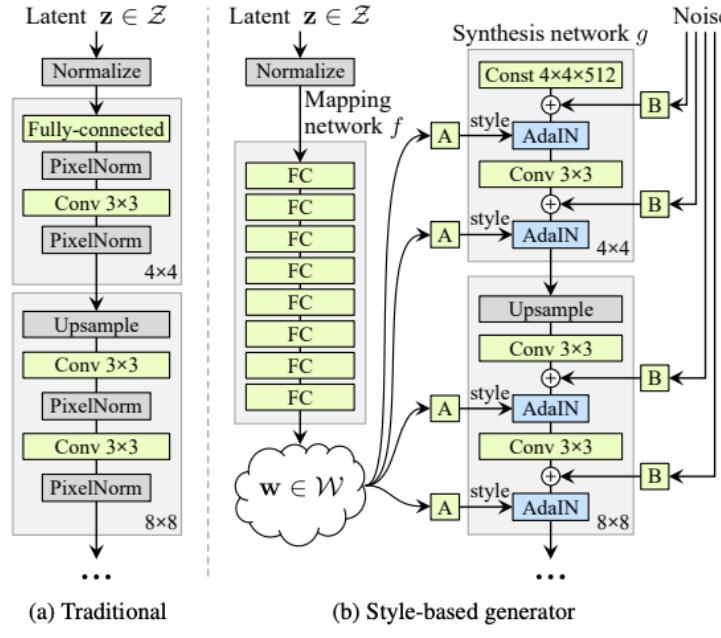
GAN과 같은 생성 모델을 통해 만들어진 이미지의 퀄리티는 많은 발전이 있었으나, 생성기에 작동 방식은 블랙 박스와 같으며 이미지 합성의 다양한 측면과 정교한 특징의 이해는 부족합니다. 그리고 latent 공간에 대한 이해 또한 부족하고 여러 생성기에서 만든 공간에 대한 보간법의 비교 방식은 존재하지 않습니다.

우리는 생성 모델을 새로운 방식으로 변경하여 이미지 합성 과정을 조절합니다. 우리의 생성기는 학습된 인풋을 사용하며 이미지의 스타일은 각 conv layer에 latent vector가 주입되어 조절합니다. 이를 통해 이미지의 변형 강도를 조절할 수 있습니다. 그리고 noise 또한 주입하여 자동적으로 미세한 특징들의 변화가 가능합니다. 판별기와 loss에 대해서는 수정하지 않습니다.

생성기는 인풋 latent code를 중간 latent 공간으로 넣습니다. 이는 네트워크의 변동성에 큰 효과를 가집니다. input latent code는 training data의 확률 밀도를 따릅니다. 그리고 이는 entanglement를 피할 수 없게 됩니다. 우리의 중간 latent 공간은 이러한 제약을 받지 않으며 disentangled하게 됩니다. latent 공간의 특징 구분을 측정하는 이전 방식은 우리에게 알맞지 않았고 이에 두가지 새로운 방식인 perceptual path length와 linear separability 방식을 제안합니다. 이 측정방식을 통해 우리는 기존 생성 모델과 우리의 생성 모델이 더욱 선형적이며 덜 entangled하다는 것을 입증합니다.

마지막으로 새로운 고품질의 사람 얼굴 데이터 셋 FFHQ를 제시합니다.

## Style-based generator



전통적인 latent code는 생성기에 제공되어집니다. 우리는 이러한 관점에서 시작을 하며 대신 학습된 상수로부터 진행을 시작합니다. latent  $z$ 를 non-linear mapping network를 통해  $w$ 를 생성합니다. 우리는 간단히  $w$ 를 512차원으로 생성하며 mapping network는 8-layers의 MLP로 구성합니다. 학습되어진 affine transformation은 이러한  $w$ 를 styles  $y = (y_s, y_b)$ 로 변경하고 이는 synthesis network  $g$ 의 convolution layer 뒤마다 있는 Adain에 주입됩니다.

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

각각의 feature map  $x_i$ 는 개별적으로 정규화가 진행되며  $y$ 에 의해 변형되어집니다. 각각의  $y$ 는  $x$  featuremap 차원의 두배입니다. 스타일 변환 방식을 생각하면 우리는 이미지 대신  $w$  vector로부터 생성된 불변 스타일  $y$ 를 사용합니다. 우리는 ‘style’이란 단어를  $y$ 에 대해서 사용합니다. 그 이유는 이미 많이 사용되어지고 있기 때문입니다. 그리고 AdaIN은 효율성과 간결한 representation으로 인해 우리의 목적에 적합합니다.

마지막으로 우리는 생성기에 노이즈를 주입하여 디테일을 생성할 수 있습니다. 이는 가우시안 노이즈와 상관없는 단일 채널 이미지로 구성되어 있으며 각 레이어마다 주입합니다.

### Quality of generated images

Method	CelebA-HQ	FFHQ
A Baseline Progressive GAN [30]	7.79	8.04
B + Tuning (incl. bilinear up/down)	6.11	5.25
C + Add mapping and styles	5.34	4.85
D + Remove traditional input	5.07	4.88
E + Add noise inputs	<b>5.06</b>	4.42
F + Mixing regularization	5.17	<b>4.40</b>

Table 1. Fréchet inception distance (FID) for various generator designs (lower is better). In this paper we calculate the FIDs using 50,000 images drawn randomly from the training set, and report the lowest distance encountered over the course of training.

우리의 생성기는 실험적으로 더 좋은 이미지를 만들어내는 것을 확인했습니다. 두 가지 데이터셋으로 실험을 진행하였습니다. A는 PGGAN이고 B는 bilinear을 통한 up/down sampling과 tuning, C는 mapping network와 AdaIN operation의 추가, 우리는 더 이상 latent code를 바로 생성기에 넣는 것은 이점이 없다는 것을 발견하고 이전의 바로 집어넣었던 방식을 제거하고 학습되어진 4\*4\*512 tensor를 사용하기 시작한 것이 D, E는 노이즈를 인풋으로 추가한 것, F는 이웃 스타일과 상관성을 없애주며 디테일한 특징을 조절 가능하게 해주는 mixing regularization을 사용한 것입니다.

우리는 우리의 방식을 WGAN-GP와 non-saturating loss 두 가지 로스를 통해 평가합니다. 이러한 방식이 최선의 결과를 보여준 것을 증명했습니다. 우리의 Contribution은 손실 함수를 수정하지 않는 것입니다. 우리는 E(generator)가 이전의 B(generator)보다 상당한 성능 증가를 얻은 것을 볼 수 있습니다.



Figure 2. Uncurated set of images produced by our style-based generator (config F) with the FFHQ dataset. Here we used a variation of the truncation trick [42, 5, 34] with  $\psi = 0.7$  for resolutions  $4^2 - 32^2$ . Please see the accompanying video for more results.

위의 이미지는 FFGQ 데이터셋이 우리의 생성기를 통해 만들어진 새로운 이미지입니다. FID score에 따라 상당히 퀄리티가 높으며 썬글라스나 다른 악세서리도 잘 합성된 것을 볼 수 있습니다. 우리는 truncation trick을 사용하여 W의 극단적인 지역에서 추출하는 것을 방지했습니다. 이는 저해상도에서만 사용되어 고해상도에서는 영향을 끼치지 않습니다.

모든 FID는 truncation trick 없이 산정되기 때문에 우리는 예시 목적으로만 사용했습니다. 모든 이미지는 1024\*1024의 해상도로 생성됩니다.

### Prior art

GAN 네트워크의 이전에는 여려개의 판별기, 다양한 해상도의 판별, self-attention등의 방식을 통해 판별기를 발전시키는데 초점을 두었습니다. 생성기 같은 경우는 input latent 공간을 정확하게 예측하거나 가우시안 모델을 통해 인풋 latent 공간을 만드는 초점을 두었습니다. 최근 C-GAN의 경우는 제네레이터의 많은 레이어를 통한 임베팅 네트워크로 class 분류기를 만들었습니다. 몇몇 저자들은 latent 값을 여러 생성기 레이어에 주입시키는 방식도 사용하였고, style modulate의 생성기는 AdaINs를 우리와 비슷하게 사용했지만, 중간 latent space나 noise를 고려하지 않았습니다.

## Properties of the style-based generator

우리의 생성기는 규모별 조정을 통해 이미지의 변형을 조절할 수 있습니다. mapping network와 affine transform은 학습된 분포에서부터 각각의 스타일에 대한 샘플을 추출하는 것으로 볼 수 있습니다. 그리고 합성 네트워크는 수집된 스타일로부터 새로운 이미지를 만드는 것입니다. 네트워크에서 각각의 스타일에 대해 규정하는 것은 이미지의 특정 부분만을 수정할 수 있게 됩니다.

정규화에 대한 이유를 보자면 AdaIN이 어떻게 작동하는지 봐야합니다. 먼저 각각의 채널에 대해 정규화를 진행하고, 스타일에 대해서 곱하고 편향을 더해줍니다. 채널별로 살펴보자면 스타일 벡터에 의해서 피쳐의 값별 상대적 중요도가 수정되어집니다. 그러나 정규화로 인해서 원래의 통계량에 의존하지 않습니다. 그래서 각각의 스타일은 AdaIN이 적용되어지기 전 한 번의 convolution으로 통제되어 집니다.

### Style mixing

스타일을 규정하기 위해서 우리는 학습 중 하나가 아닌 두개의 latent를 이용하는 mixing 규제를 사용한다. 이미지를 생성하기 위해 우리는 간단히 하나의 latent를 다른 하나로 전환한다. 우리는 이것을 style mixing이라 하며 랜덤적으로 전환하여 사용한다. 제대로 말하자면, 우리는 두 개의 latent  $z_1, z_2$ 를 mapping network를 통해 생성하고 해당하는  $w_1, w_2$ 를 가진다.  $w_1$ 과  $w_2$ 가 번갈아가면서 들어가 규제가 이루어지고 이는 스타일 관연관성을 예방한다.

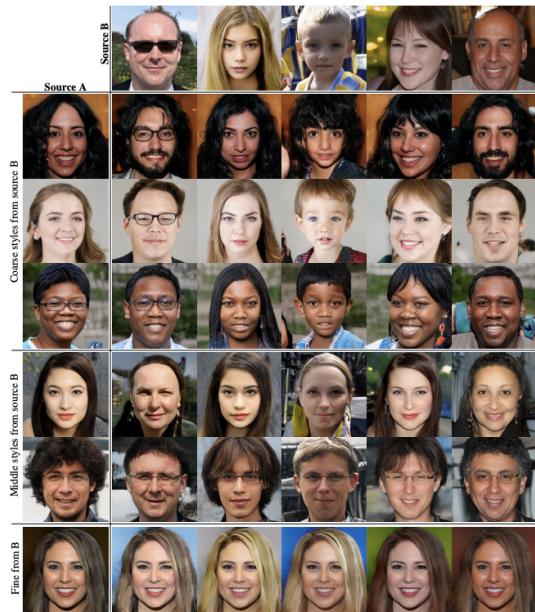


Figure 3. Two sets of images were generated from their respective latent codes (sources A and B); the rest of the images were generated by copying a specified subset of styles from source B and taking the rest from source A. Copying the styles corresponding to coarse spatial resolutions ( $4^2 - 8^2$ ) brings high-level aspects such as pose, general hair style, face shape, and eyeglasses from source B, while all colors (eyes, hair, lighting) and finer facial features resemble A. If we instead copy the styles of middle resolutions ( $16^2 - 32^2$ ) from B, we inherit smaller scale facial features, hair style, eyes open/closed from B, while the pose, general face shape, and eyeglasses from A are preserved. Finally, copying the fine styles ( $64^2 - 1024^2$ ) from B brings mainly the color scheme and microstructure.

이는 두 가지 이미지의 스타일이 섞인 결과이다. 스타일의 변형 정도를 조절하면서.

Mixing regularization	Number of latents during testing			
	1	2	3	4
E 0%	4.42	8.22	12.88	17.41
50%	4.41	6.10	8.71	11.61
F 90%	<b>4.40</b>	<b>5.11</b>	6.88	9.03
100%	4.83	5.17	<b>6.63</b>	<b>8.40</b>

Table 2. FIDs in FFHQ for networks trained by enabling the mixing regularization for different percentage of training examples. Here we stress test the trained networks by randomizing 1...4 latents and the crossover points between them. Mixing regularization improves the tolerance to these adverse operations significantly. Labels E and F refer to the configurations in Table 1.

multiple latents를 사용하여 더 높은 FID score를 기록한 것을 볼 수 있다.

### Stochastic variation

인물 사진에는 확률적으로 많은 측면이 있다. 예를 들어 수염, 주근깨, 모공, 머리카락의 위치 등등. 이러한 모든 것은 올바른 분포를 따르는 한 랜덤적으로 될 수 있다. 기존의 생성기가 확률적 변동을 어떻게 실행시켰는지 보자. 오직 인풋이 네트워크를 통하여 이 네트워크가 위와 같은 랜덤적으로 생성하기 위해 새로운 길을 발견해야 한다. 그러나 이러한 것은 네트워크의 용량을 소비할 뿐더러 정밀한 신호를 해치는 경우가 생긴다. 그래서 우리는 각각의 conv layer 이후 noise를 주입하도록 한다.



(a) Generated image (b) Stochastic variation (c) Standard deviation

Figure 4. Examples of stochastic variation. (a) Two generated images. (b) Zoom-in with different realizations of input noise. While the overall appearance is almost identical, individual hairs are placed very differently. (c) Standard deviation of each pixel over 100 different realizations, highlighting which parts of the images are affected by the noise. The main areas are the hair, silhouettes, and parts of background, but there is also interesting stochastic variation in the eye reflections. Global aspects such as identity and pose are unaffected by stochastic variation.

우리는 위의 사진을 통해서 노이즈가 정교한 측면에서 영향을 주는 것을 확인할 수 있다. (a)는 생성한 이미지, (b)는 정교한 부분의 노이즈에 따른 이미지 변형, (c)는 노이즈가 영향을 주는 부분을 하이라이트화 시킨 것이다.

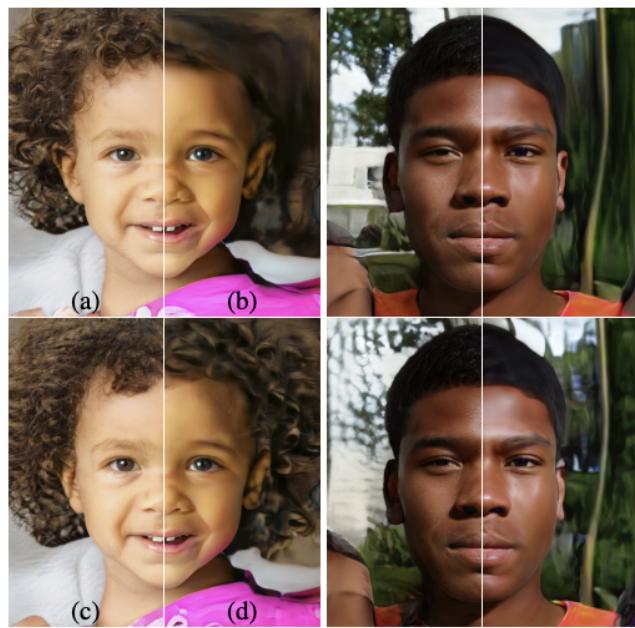


Figure 5. Effect of noise inputs at different layers of our generator. (a) Noise is applied to all layers. (b) No noise. (c) Noise in fine layers only ( $64^2 - 1024^2$ ). (d) Noise in coarse layers only ( $4^2 - 32^2$ ). We can see that the artificial omission of noise leads to featureless “painterly” look. Coarse noise causes large-scale curling of hair and appearance of larger background features, while the fine noise brings out the finer curls of hair, finer background detail, and skin pores.

위의 사진은 노이즈가 어디 부분 레이어에 주입되었는지에 따른 영향을 나타내고 있다. (a)는 모든 레이어, (b)는 no noise, (c)는 fine layers, (d)는 coarse layer에 노이즈가 주입된 결과이다. 우리는 이러한 노이즈가 원래 샘플에서 비슷한 유형의 샘플을 만들어내는 것을 확인하였고 이제는 더 이상 초기 액티베이션에서 확률적 효과를 생성할 필요가 없어졌습니다.

### Separation of global effects from stochasticity

이전에서 살펴봤듯이 스타일은 큰 이펙트 포즈나 정체성 등 그리고 노이즈는 디테일한 부분에 대해 변경해줍니다. 이러한 관측은 고정적인 통계량은 안정적인 이미지를 표현하고 변경되는 피쳐는 특정적인 변형을 일으킵니다. 우리의 스타일 기반 생성기에서 스타일은 전체이미지에 영향을 끼치게 됩니다. 그리고 노이즈는 세밀한 표현을 변형시킵니다.

## Disentanglement studies

특정 구분에 대해서는 다양한 정의가 있습니다. 그러나 대부분의 목표는 latent 공간이 선형적으로 표현이 되길 원하며 하나의 요인으로 통제가 되길 바랍니다. 그러나  $Z$ 의 각 샘플링 확률은 학습 데이터와 일치해야 합니다.

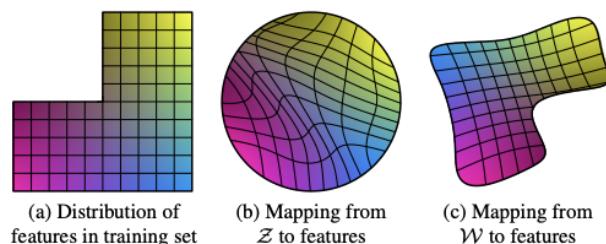


Figure 6. Illustrative example with two factors of variation (image features, e.g., masculinity and hair length). (a) An example training set where some combination (e.g., long haired males) is missing. (b) This forces the mapping from  $\mathcal{Z}$  to image features to become curved so that the forbidden combination disappears in  $\mathcal{Z}$  to prevent the sampling of invalid combinations. (c) The learned mapping from  $\mathcal{Z}$  to  $\mathcal{W}$  is able to “undo” much of the warping.

우리의 생성기의 가장 큰 이점은 중간 latent space  $W$ 가 고정 분포에 따른 샘플링을 하지 않는다는 것입니다.  $W$ 는 학습된  $Z$ 에 의해 샘플링되어집니다. 이러한 것은 좀 더 선형적으로 (c)와 같은 매핑을 하게끔 해줍니다. 우리는 이러한 구분가능한 representation에서 sample을 만드는 것이 더 쉽다고 가정을 합니다. 따라서 덜 구분가능한  $W$ 는 변동의 원인을 알 수 없습니다.

아쉽게도 이러한 특징 구분에 대한 우리 모델에 적합한 측정 지표는 딱히 있지 않았습니다. 따라서 우리는 이를 가능케 하기 위한 두 가지 방식을 제시합니다.

### Perceptual path length

latent 공간 벡터의 보간은 많은 비선형성 변화를 발생시켰습니다. 두 벡터를 보간할 때 얼마나 급격하게 이미지가 바뀌는지, 서서히 바뀔 경우 이를 disentanglement하다고 합니다.

### Linear serability

latent space에서 특성이 얼마나 선형적으로 분류될 수 있는가에 대한 평가 지표.

## Conclusion

우리의 여러 결과를 통해 스타일에 관해서는 일반적인 GAN 모델은 Style-Gan보다 부족한 것을 확인할 수 있었습니다. 이러한 것은 평가 지표에서 나타났으며 스타일 변화의 큰 부분부터 디테일까지에서도 보입니다. 그리고 중간 latent space의 효과까지 입증되었습니다.