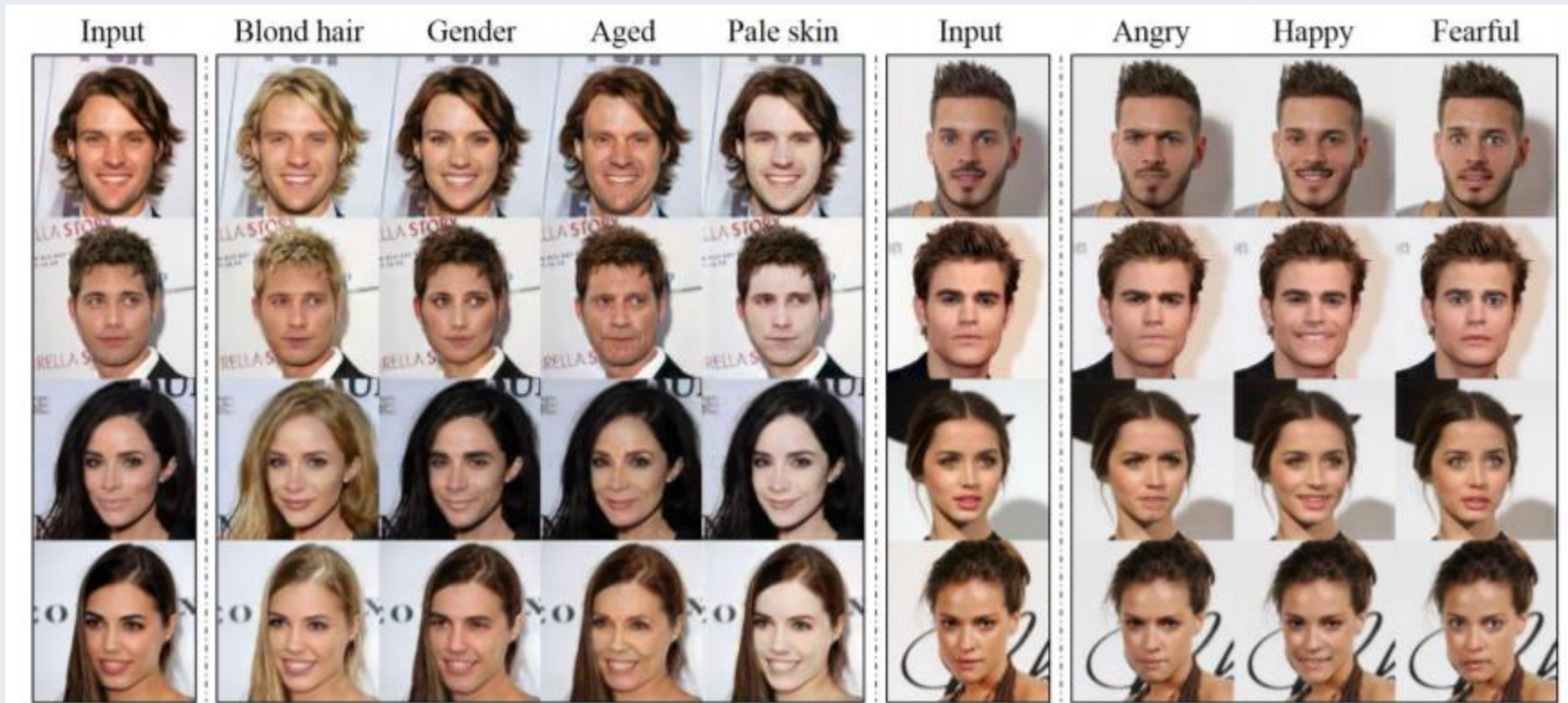


# 생성적 적대 신경망(GAN)



# GAN (Generative Adversarial Network)

GAN은 적대 관계를 가지고 서로 경쟁하는 2개의 딥러닝 모델을 사용합니다.

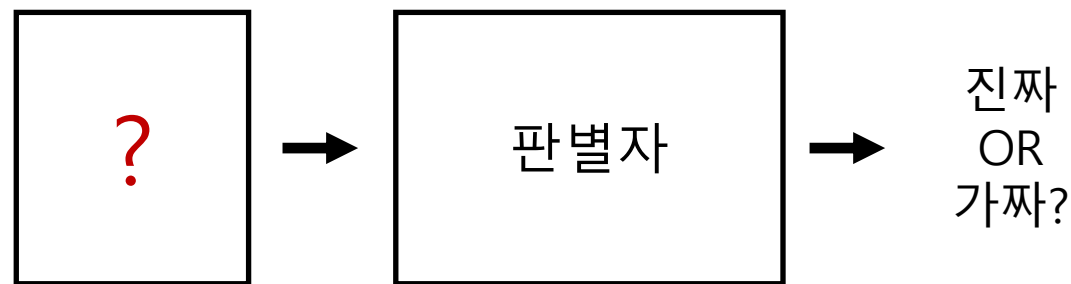
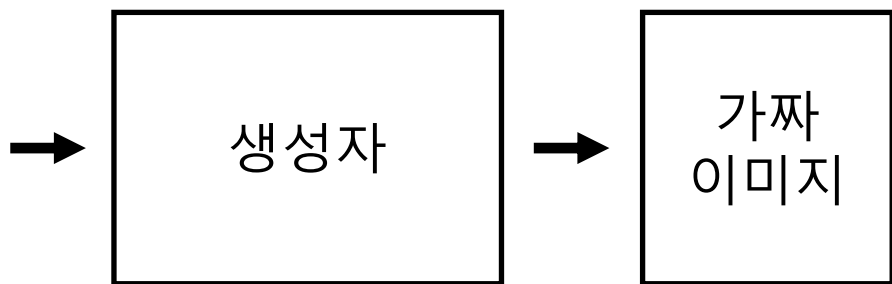
생성자는 위조 이미지를 만들고 판별자는 위조 이미지와 진짜 이미지를 구분합니다..

훈련이 계속됨(적대 상호 작용)에 따라 판별자뿐만 아니라 사람이 보아도 그럴싸한 가짜를 만드는 생성자가 됩니다.

생성자(Generator)

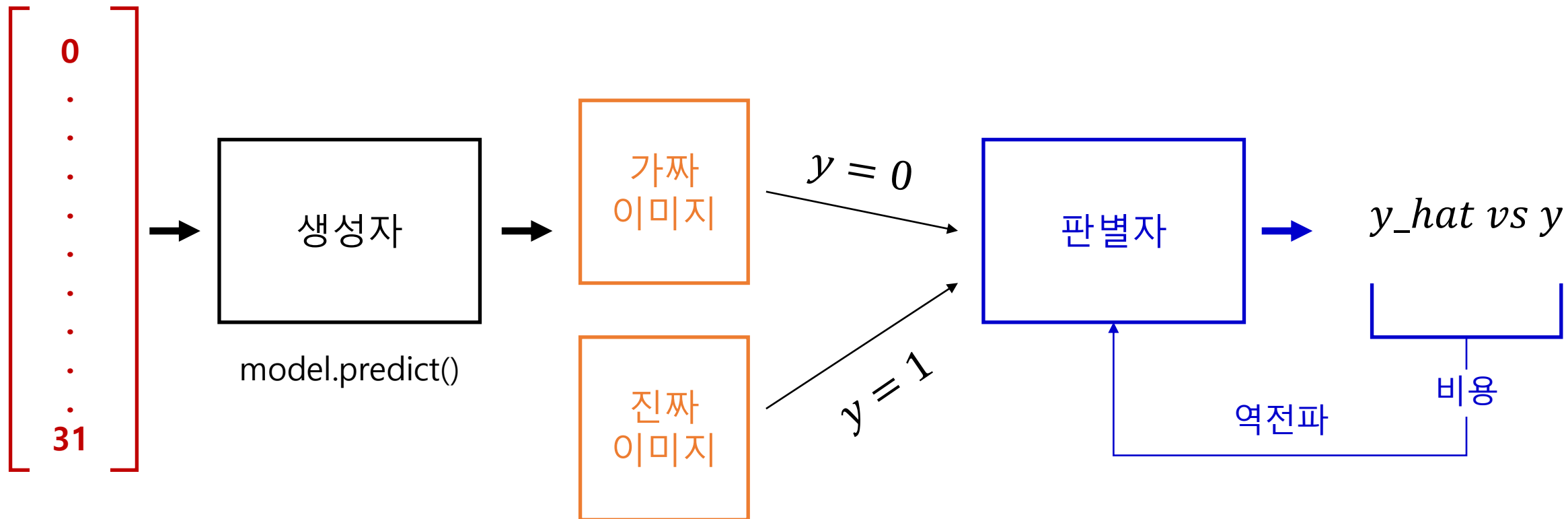
판별자(Discriminator)

$\begin{bmatrix} 0 \\ \vdots \\ 31 \end{bmatrix}$



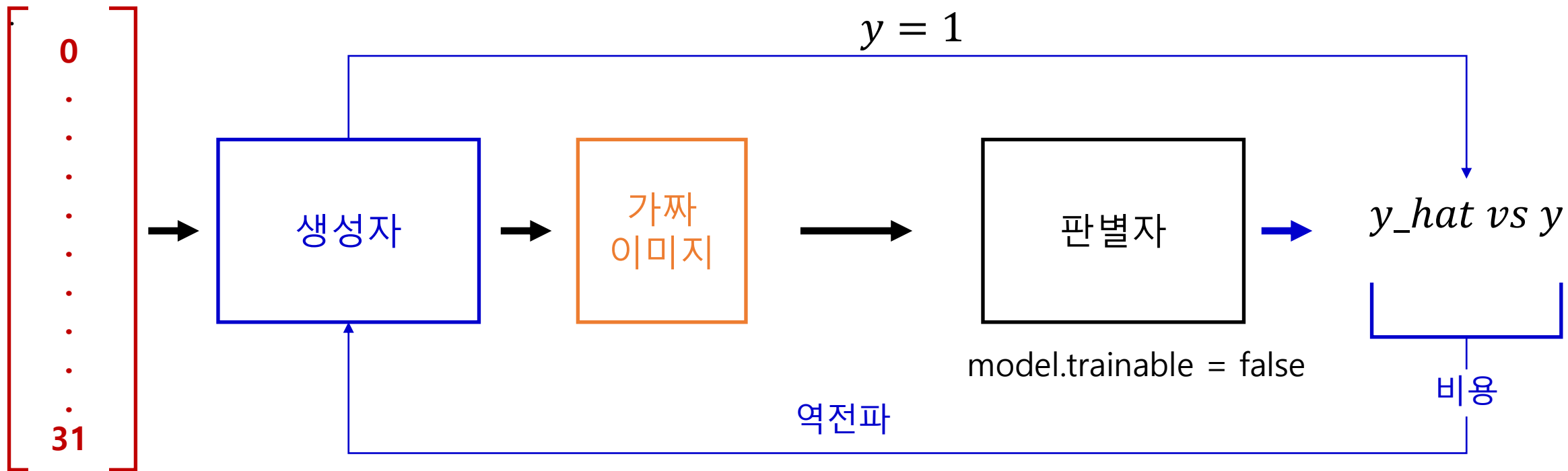
# 판별자 훈련

- 생성자의 정방향 계산으로 가짜 이미지를 만듭니다.
- 생성된 가짜 이미지와 진짜 이미지를 섞어 배치를 만들고, 레이블을 더해 판별자를 훈련합니다.
- 판별자는 이미지가 진짜일 가능성에 해당하는 예측( $y_{\hat{}}$ )을 출력합니다.
- 판별자의 예측  $y_{\hat{}}$  과 진짜 레이블  $y$ 에 대한 손실을 계산합니다.
- 역전파로 판별자의 파라미터를 튜닝합니다.



# 생성자 훈련

- 생성자는 랜덤한 잡음 벡터  $z$ 를 입력으로 받고 가짜 이미지를 만듭니다.
- 생성자가 만든 가짜 이미지를 판별자로 주입합니다. 가짜 이미지의 레이블을 모두 진짜( $y=1$ )로 입력합니다.
- 판별자는 입력 이미지가 진짜인지 가짜인지에 대한 예측  $y_{\hat{}}$ 을 출력합니다.
- 크로스-엔트로피 손실을 사용해 생성망 신경망의 파라미터를 튜닝합니다.
- 생성자는 판별자가 속을 만한 가짜 이미지를 만드는 방법을 학습합니다.
- 이 비용을 최소화하면 판별자가 진짜로 잘못 판단할 가짜 이미지를 만들도록 생성자가 학습이 될 것입니다.



# GAN Lab

<https://poloclub.github.io/ganlab/>

Play with Generative Adversarial Networks (GANs) in your browser!

Fork us on GitHub



## GAN Lab

Data Distribution



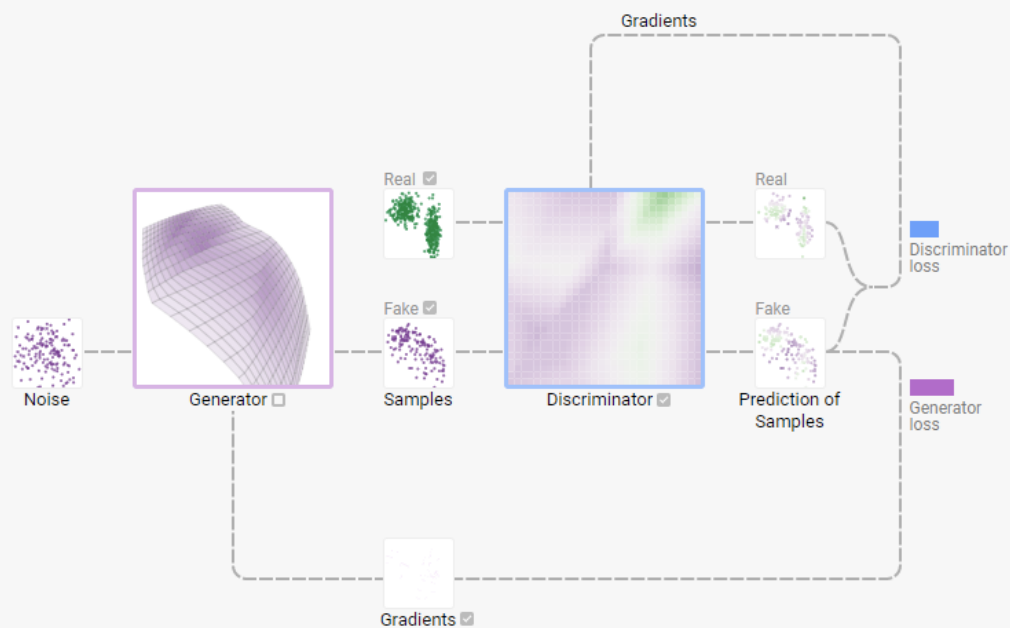
☒ Use pre-trained model



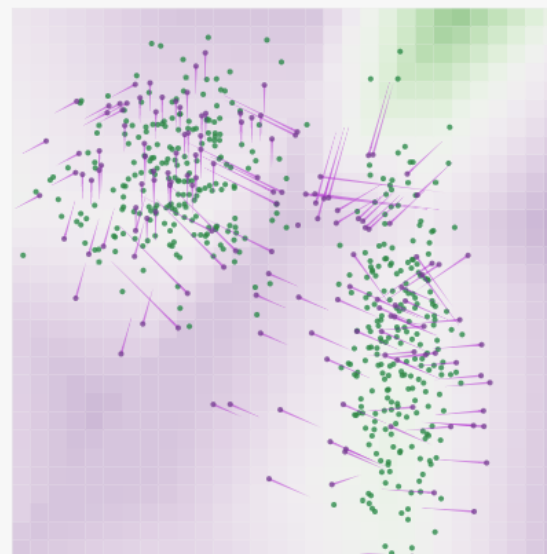
Epoch

001,931

MODEL OVERVIEW GRAPH



LAYERED DISTRIBUTIONS



Each dot is a 2D data sample: **real samples**; **fake samples**.

Background colors of grid cells represent **discriminator's** classifications. Samples in **green regions** are likely to be real; those in **purple regions** likely fake.

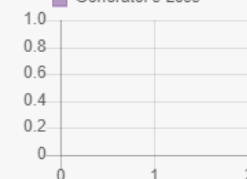
**Manifold** represents **generator's** transformation results from noise space. Opacity encodes density: darker purple means more samples in smaller area.

Pink lines from fake samples represent **gradients** for generator.

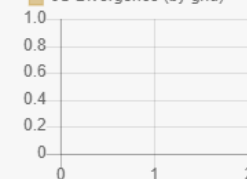
This sample needs to move upper right to decrease generator's loss.

METRICS

Discriminator's Loss  
 Generator's Loss

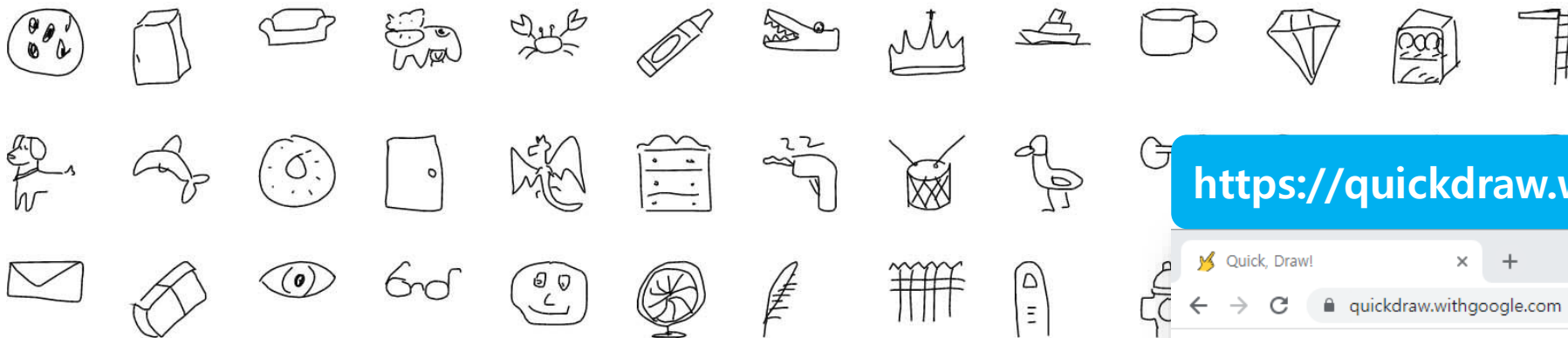


KL Divergence (by grid)  
 JS Divergence (by grid)



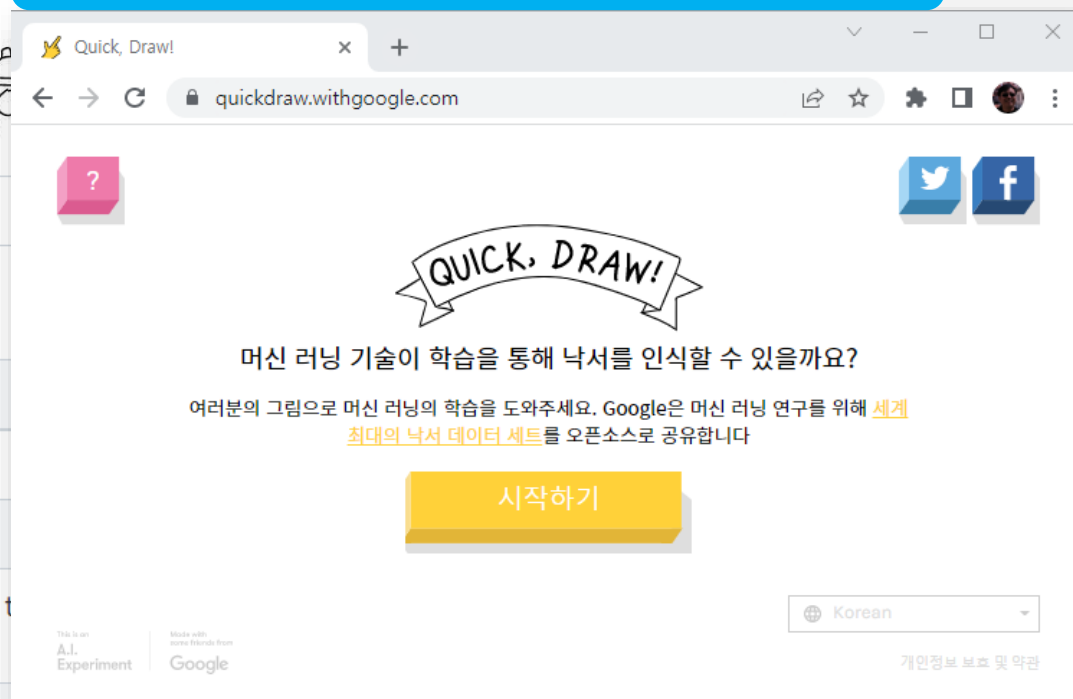
# GAN 구현 실습 - Quick draw

<https://github.com/googlecreativelab/quickdraw-dataset>



<https://quickdraw.withgoogle.com/>

Key	Type	Description
key_id	64-bit unsigned integer	A unique identifier across all drawings.
word	string	Category the player was prompted to draw.
recognized	boolean	Whether the word was recognized by the game.
timestamp	datetime	When the drawing was created.
countrycode	string	A two letter country code (ISO 3166-1 alpha-2) of where the drawing was located.
drawing	string	A JSON array representing the vector drawing



# GAN 구현 실습 - Quick draw



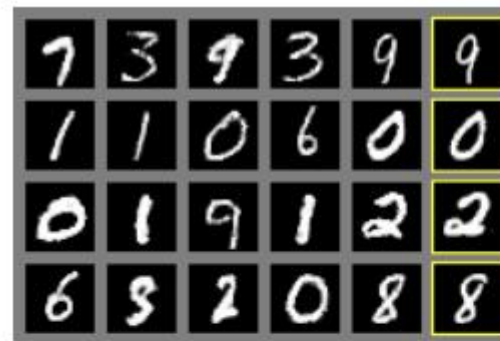
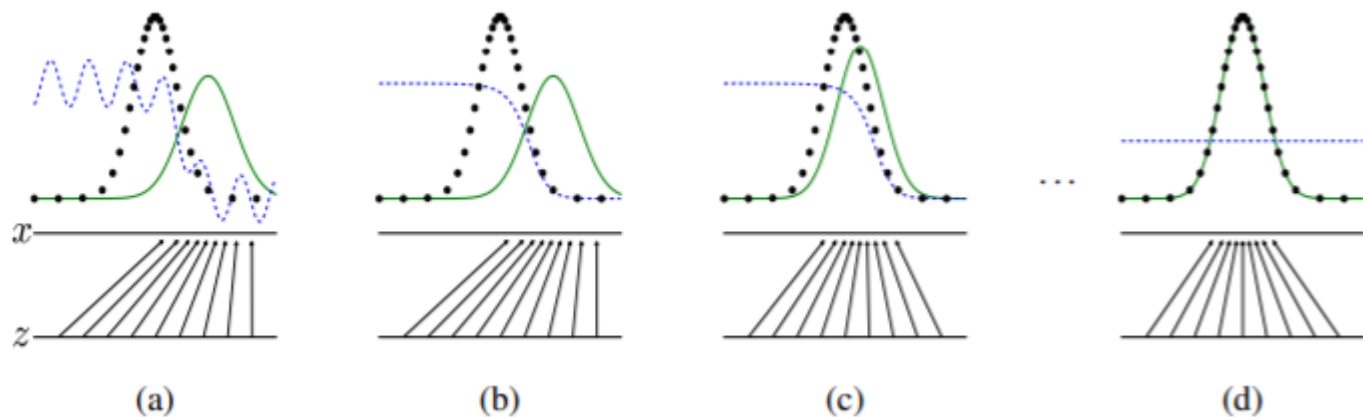
`generative_adversarial_network.ipynb`



# GAN 모델

## GAN : Generative Adversarial Nets [NIPS 2015]

- **Ian Goodfellow가 최초로 GAN (Generative Adversarial Nets)를 제안한 논문.**
- 새로운 이미지를 생성하는 생성자 Generator와 샘플 데이터와 생성자가 생성한 이미지를 구분하는 구별자 Discriminator 두 개의 네트워크 구조를 제안함.
- 생성자는 구별자를 속이면서 이미지를 잘 생성하려고 하며, 구별자는 주어진 이미지와 진짜인지 가짜인지 판별함.



a)



b)



c)



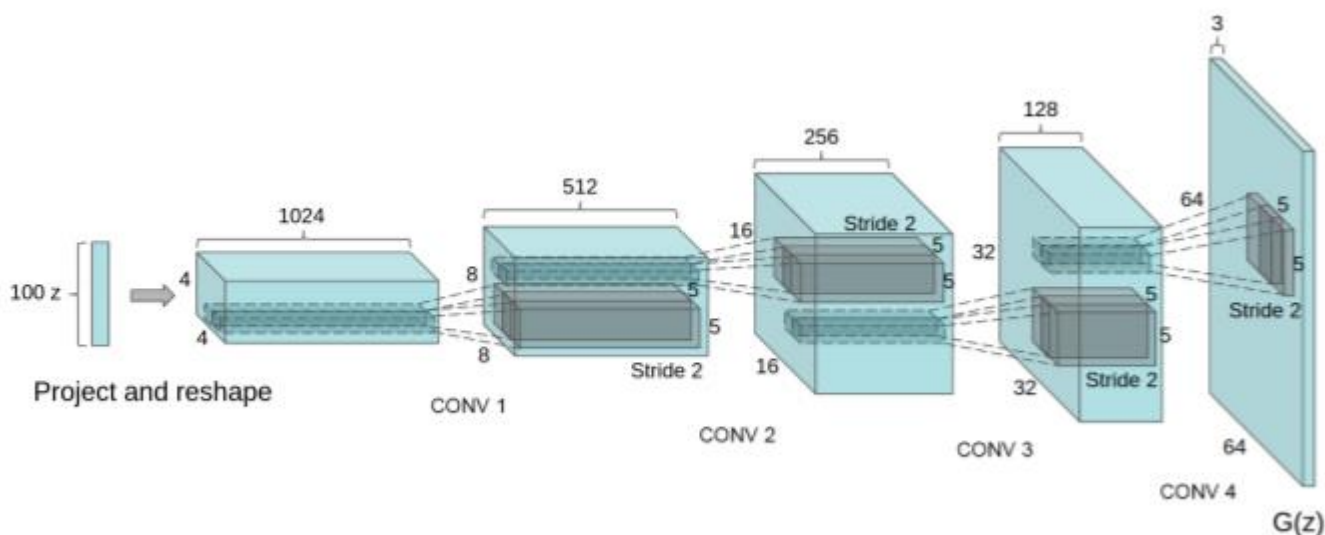
d)



# GAN 모델

## DCGAN : Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks [ICLR 2016]

- 기존의 **GAN에 CNN의 네트워크를 도입**한 DCGAN (Deep Convolutional Generative Adversarial Networks)를 제안함.
- Supervised learning에서 CNN이 큰 역할을 하고 있는데, unsupervised learning에서는 CNN이 주목을 덜 받고 있었음.
- 이렇게 CNN에서 성공적인 점을 GAN에도 적용하여 기존 GAN보다 훨씬 좋은 성능을 내게 되었음.
- 그 전까지 GAN만 사용하면 성능이 좋지 않았으나, **DCGAN 이후로부터 GAN의 발전이 많이 되었음.**



# GAN 모델

## pix2pix, CycleGAN : Image to Image Translation [CVPR 2017]

- 이미지 데이터셋을 사용하여 input 이미지와 output 이미지를 mapping하는 것을 목표로 하는 생성모델
- pix2pix 모델 : Paired Image-to-Image Translation
- CycleGAN 모델 : Unpaired Image-to-Image Translation

<https://phillipi.github.io/pix2pix/>

Paired Image-to-Image Translation

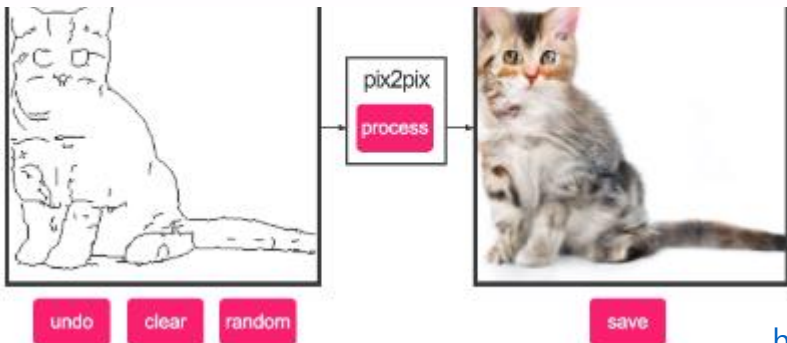
Day to night



Edges to photo



Available from: <https://arxiv.org/abs/1611.07004>



<https://www.tensorflow.org/tutorials/generative/cyclegan?hl=ko>

<https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix>



<https://velog.io/@tobigs-gm1/Image-to-Image-Translation>



# GAN 모델

## SRGAN : Super-resolution generative adversarial network [CVPR 2017]

- 이미지를 고해상도로 만들기 위한 GAN인 SRGAN(Super-resolution generative adversarial network)은 4배 확대 요소에 대해서 사실적이게 자연스러운 이미지 추론할 수 있는 최초의 프레임워크임
- Adversarial loss와 content loss로 구성된 perceptual loss function 을 제안함
- Adversarial loss는 초 해상도 이미지와 원본 이미지를 구별하도록 훈련 된 판별기(discriminator) 네트워크를 사용하여 솔루션을 자연스러운 이미지로 만들수 있도록 함
- 픽셀 공간의 유사성 대신 perceptual 유사성에 의해 유발 된 content loss를 사용함



Figure 2: From left to right: bicubic interpolation, deep residual network optimized for MSE, deep residual generative adversarial network optimized for a loss more sensitive to human perception, original HR image. Corresponding PSNR and SSIM are shown in brackets. [4× upscaling]

# GAN 모델

## StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation [CVPR 2018]

- 이 논문에서는 하나의 신경망을 이용해서 **다중 도메인을 가진 이미지 변환**을 할 수 있는 방법을 제시함
- 기존 연구[CycleGAN, DiscoGAN, pix2pix, cGAN 등]에서는 한 개의 특징만을 학습해서 변환하는 방법을 제시하였으며, 이 방법으로 하나의 도메인으로만 변환 가능하였음

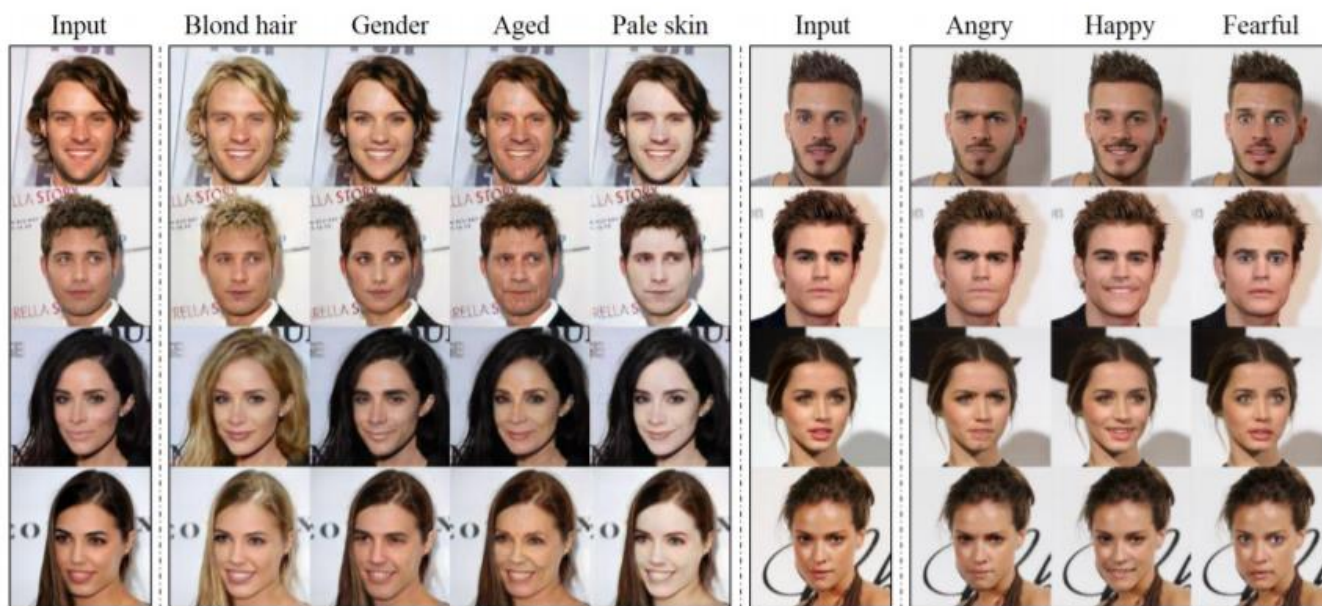


Figure 1. Multi-domain image-to-image translation results on the CelebA dataset via transferring knowledge learned from the RaFD dataset. The first and sixth columns show input images while the remaining columns are images generated by StarGAN. Note that the images are generated by a single generator network, and facial expression labels such as angry, happy, and fearful are from RaFD, not CelebA.

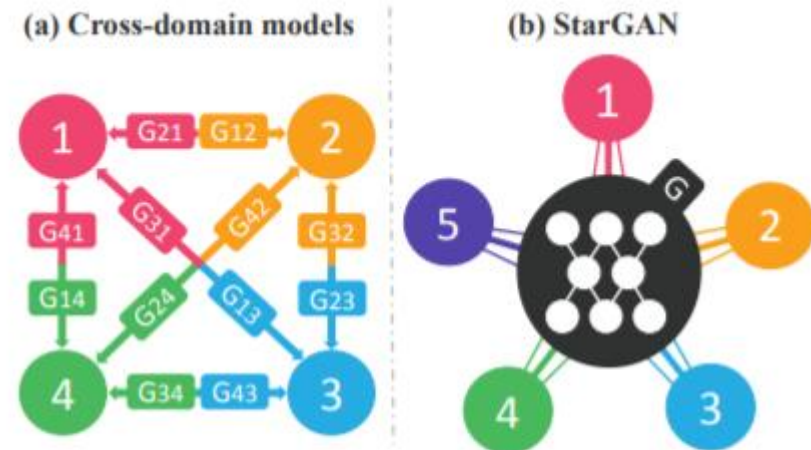


Figure 2. Comparison between cross-domain models and our proposed model, StarGAN. (a) To handle multiple domains, cross-domain models should be built for every pair of image domains. (b) StarGAN is capable of learning mappings among multiple domains using a single generator. The figure represents a star topology connecting multi-domains.

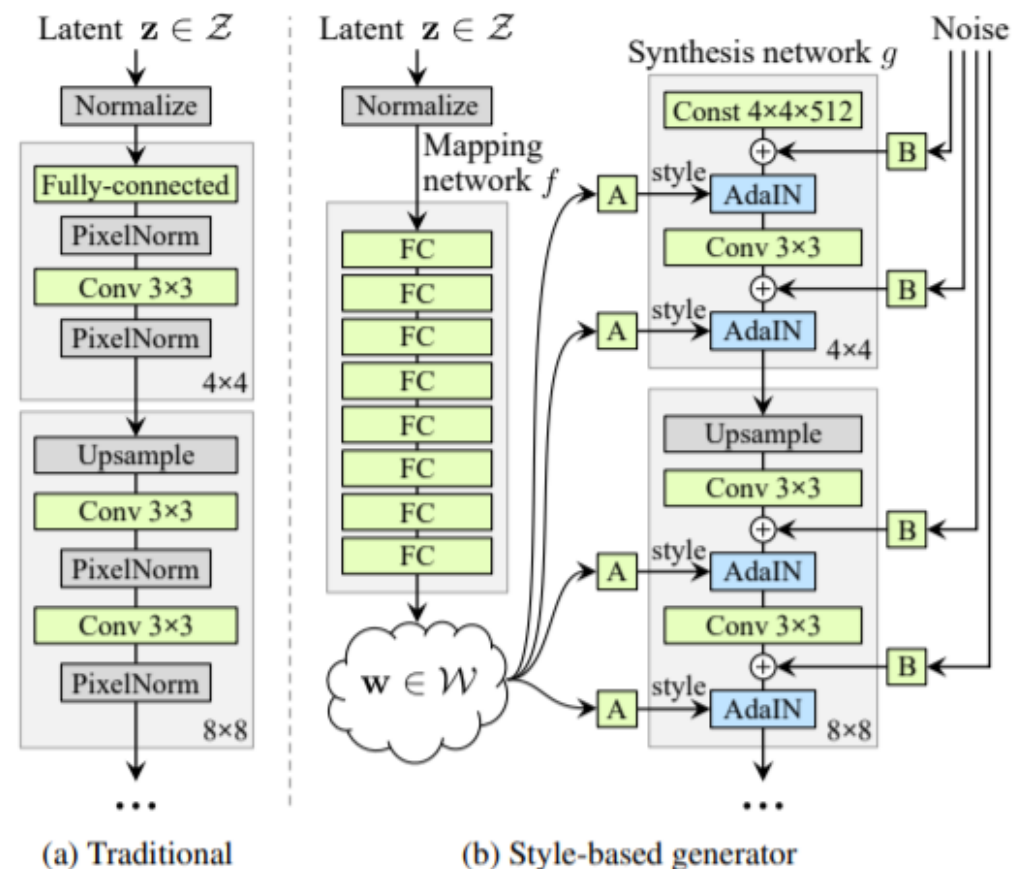
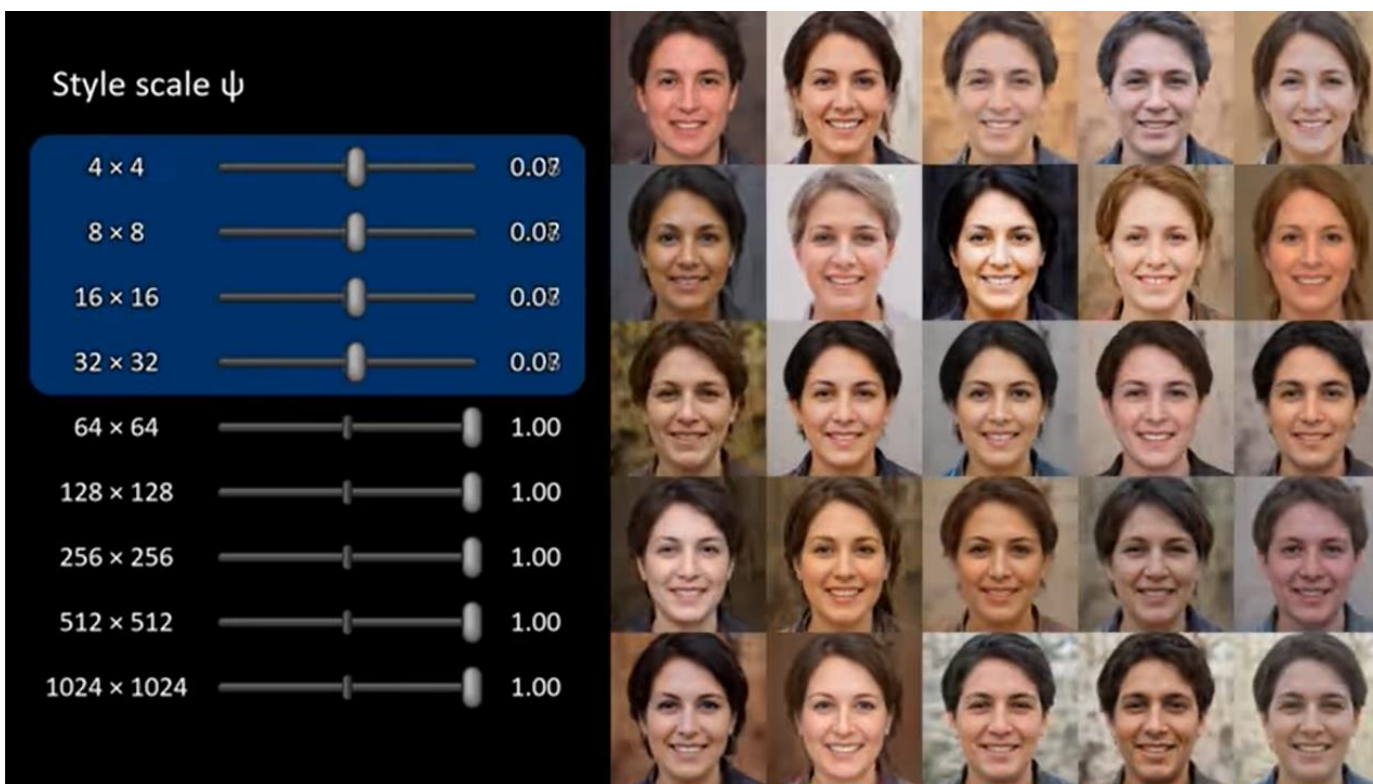


# GAN 모델

## StyleGAN : Style-Based Generator Architecture for Generative Adversarial Networks [CVPR 2019]

- Generator 네트워크에서 각 레이어마다 스타일 정보를 입히는 방식으로 학습함
- 이미지의 스타일 (성별, 포즈, 머리색, 피부톤 등)을 변경할 수 있음

<https://github.com/NVlabs/stylegan>



# DCGAN (Deep Convolutional Generative Adversarial Networks) 구현 실습



<https://www.tensorflow.org/tutorials/generative/dcgan?hl=ko>

THANK YOU

[kgpark88@gmail.com](mailto:kgpark88@gmail.com)