Unsupervised Cross-Domain Image Generation

Yunjey Choi

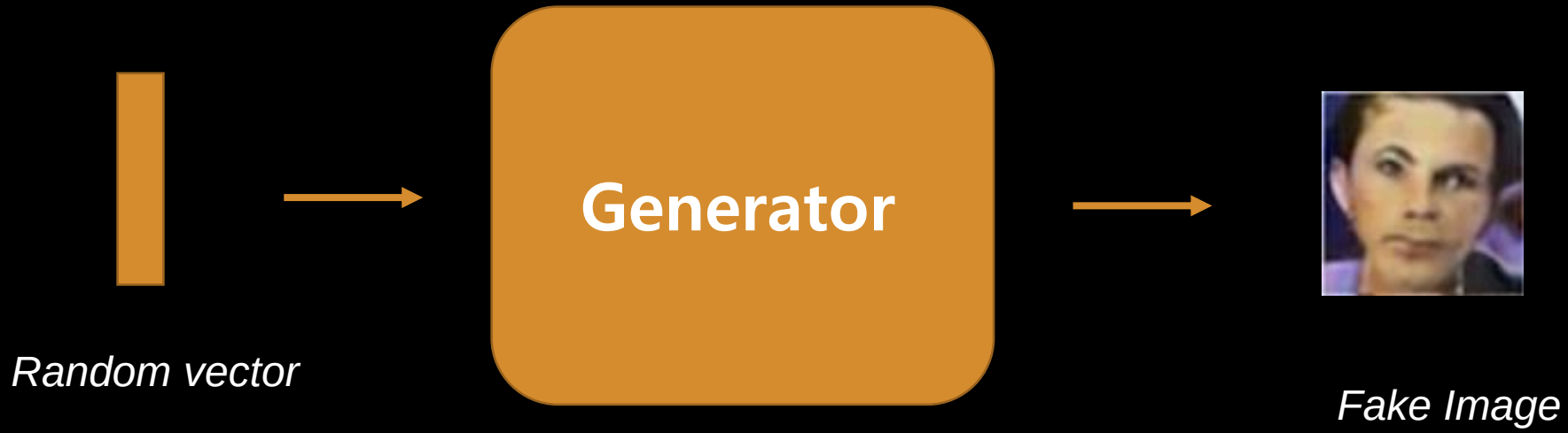
Deep Learing Study

2017.1.9

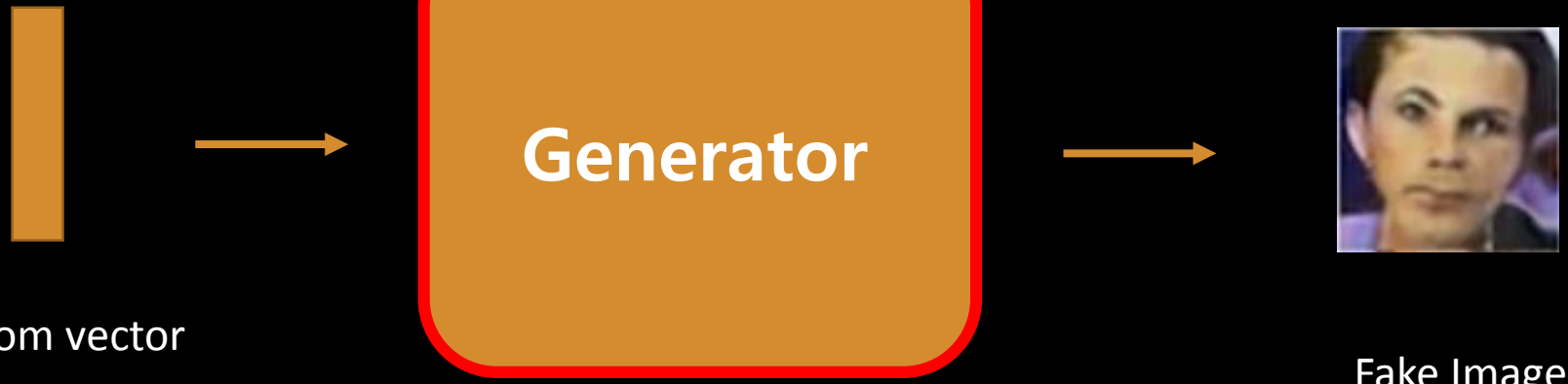
Table of Contents

1. GAN

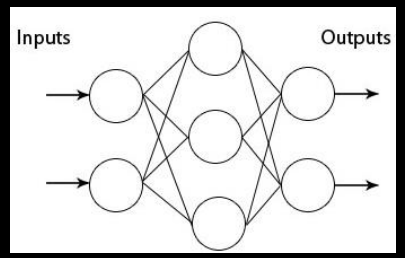
Generative Adversarial Network (GAN)



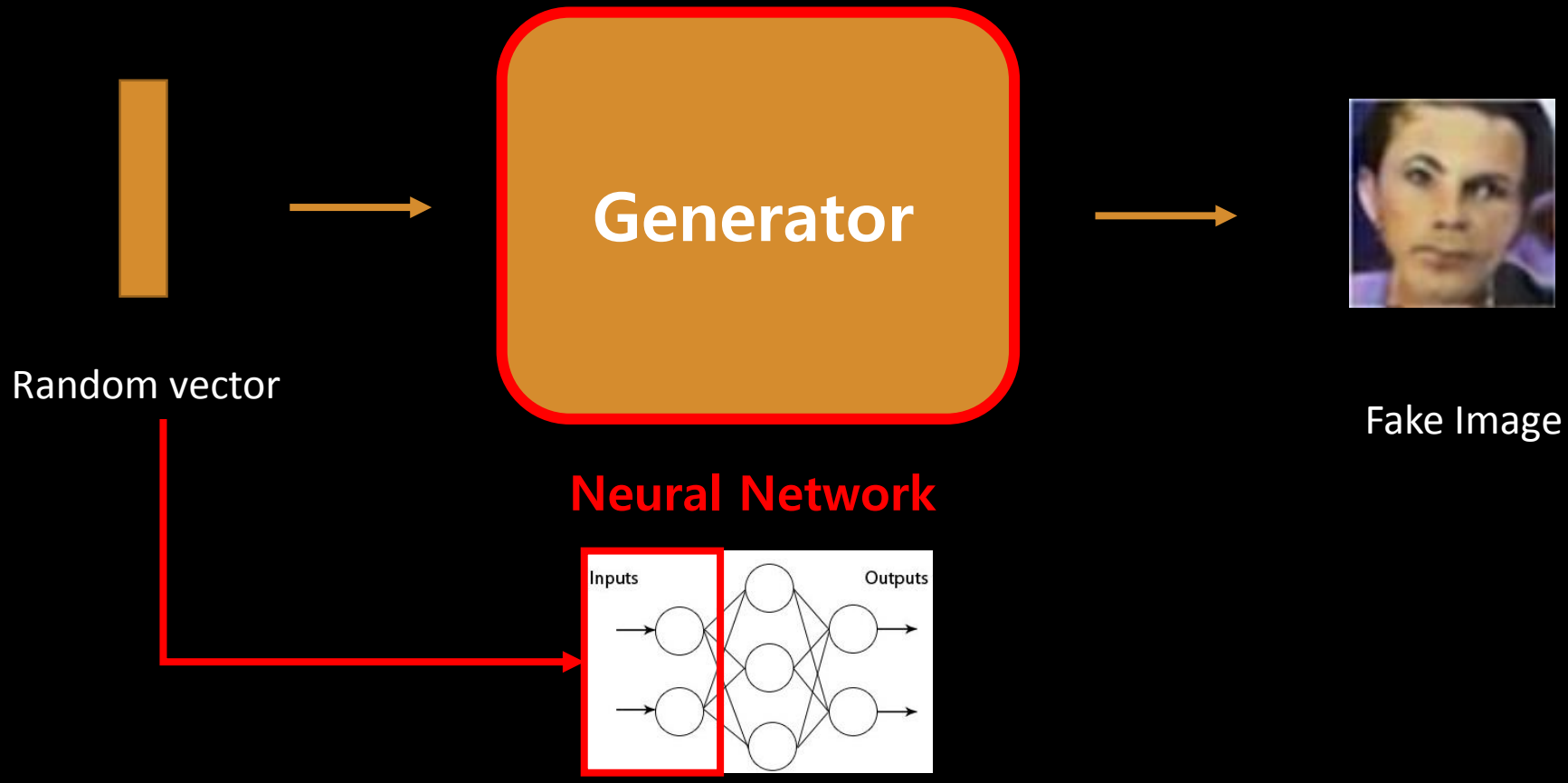
Generative Adversarial Network (GAN)



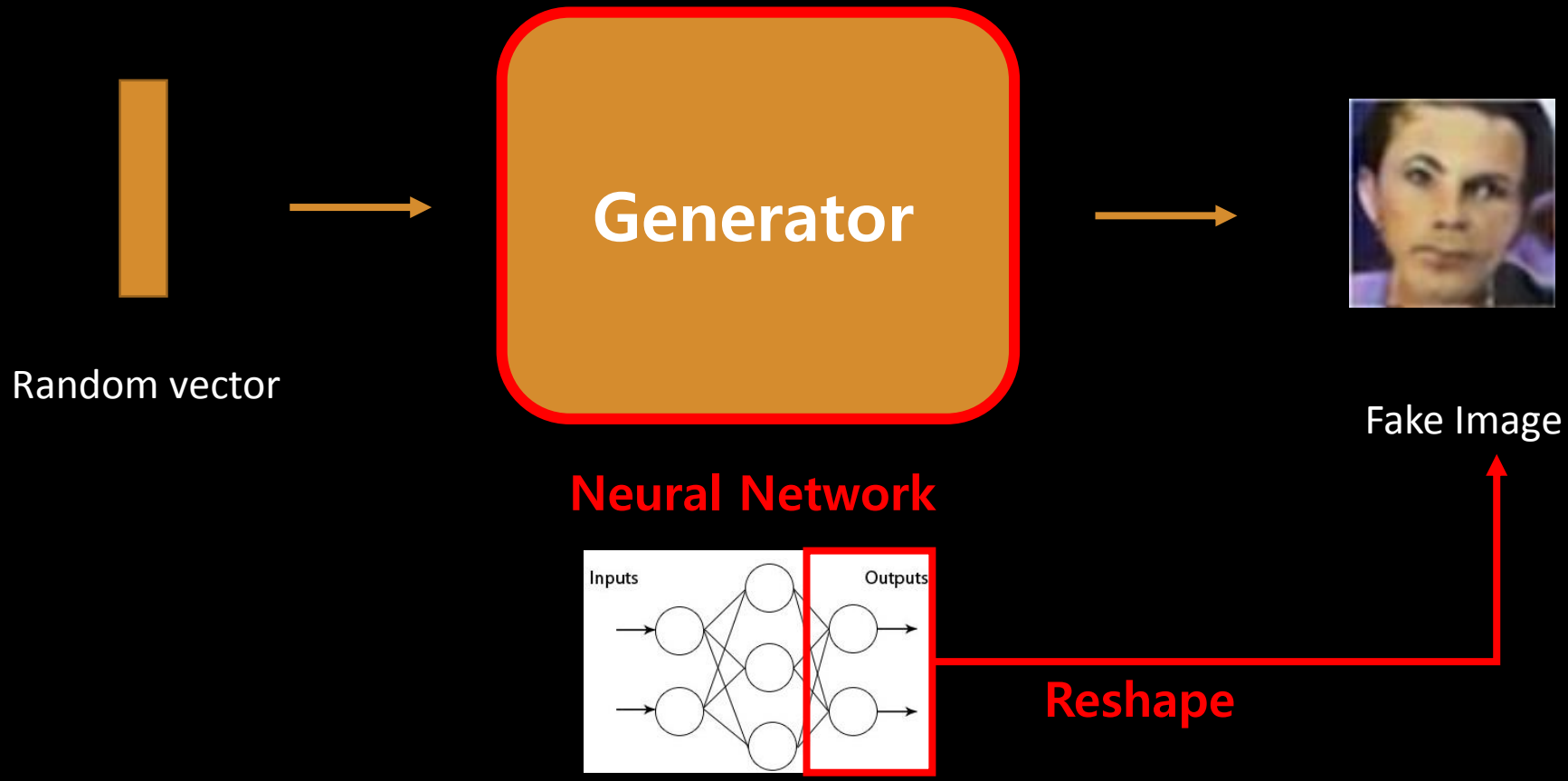
Neural Network



Generative Adversarial Network (GAN)

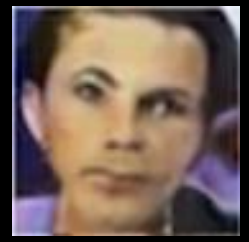


Generative Adversarial Network (GAN)



Generative Adversarial Network (GAN)

Real Image



Fake Image

진짜 이미지인가?

Discriminator

1 (Yes)

0 (No)

Generative Adversarial Network (GAN)

진짜 이미지인가?

Real Image

Discriminator

1 (Yes)

식별자는 진짜 이미지를
'진짜'로 판별하도록 학습

Generative Adversarial Network (GAN)

Real Image

진짜 이미지인가?

Discriminator

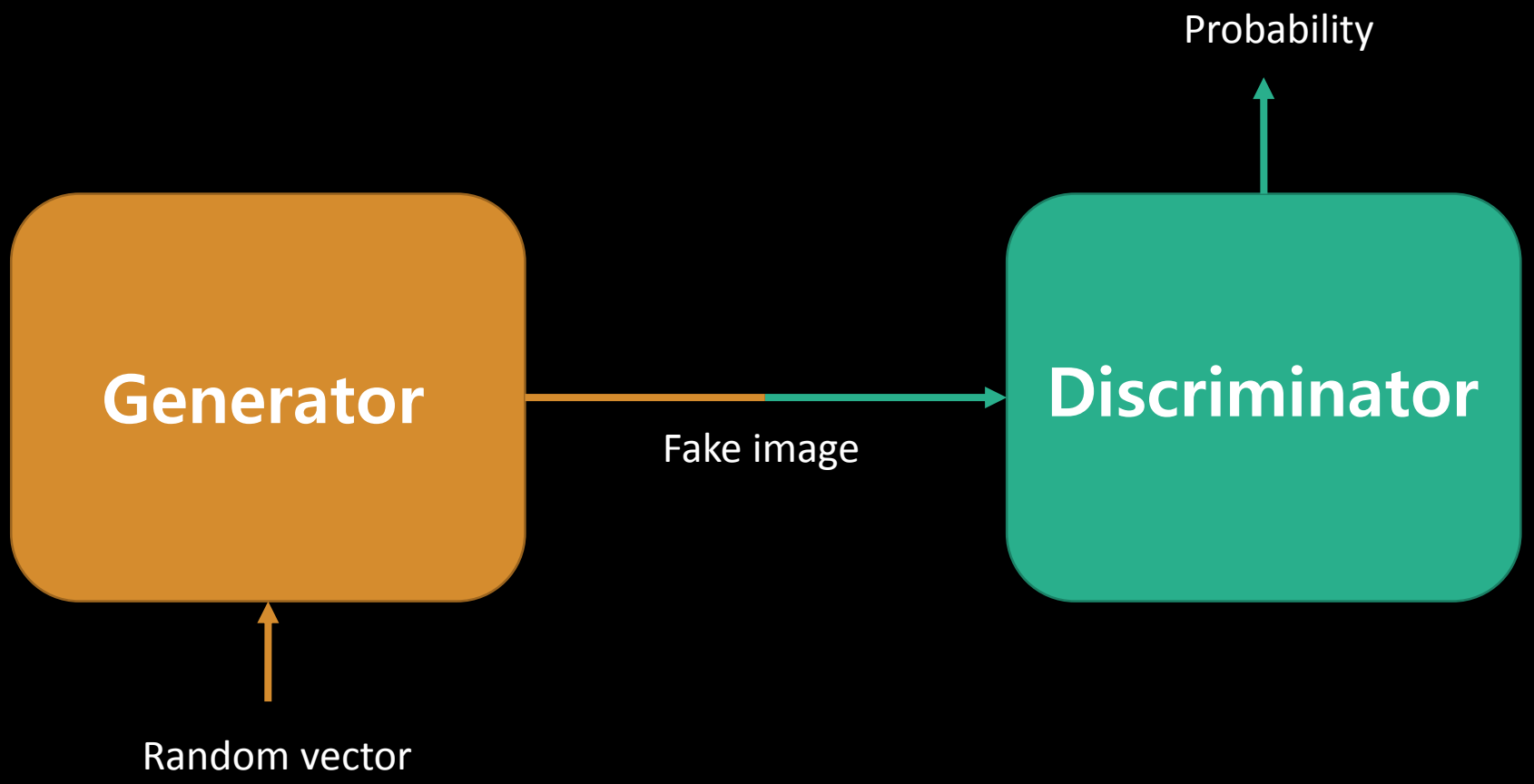
1 (Yes)

0 (No)

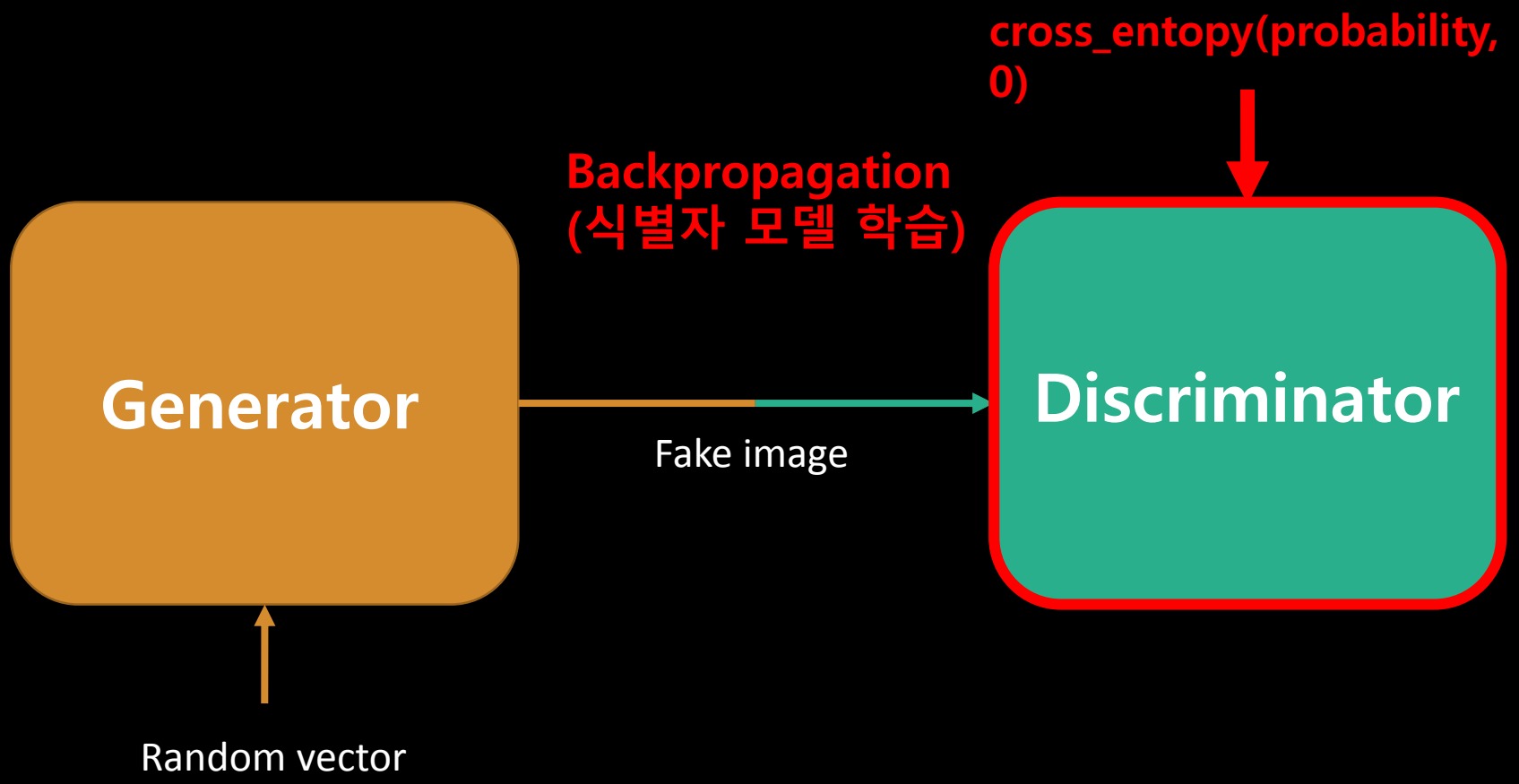
Generator가 생성한 이미지

Fake Image

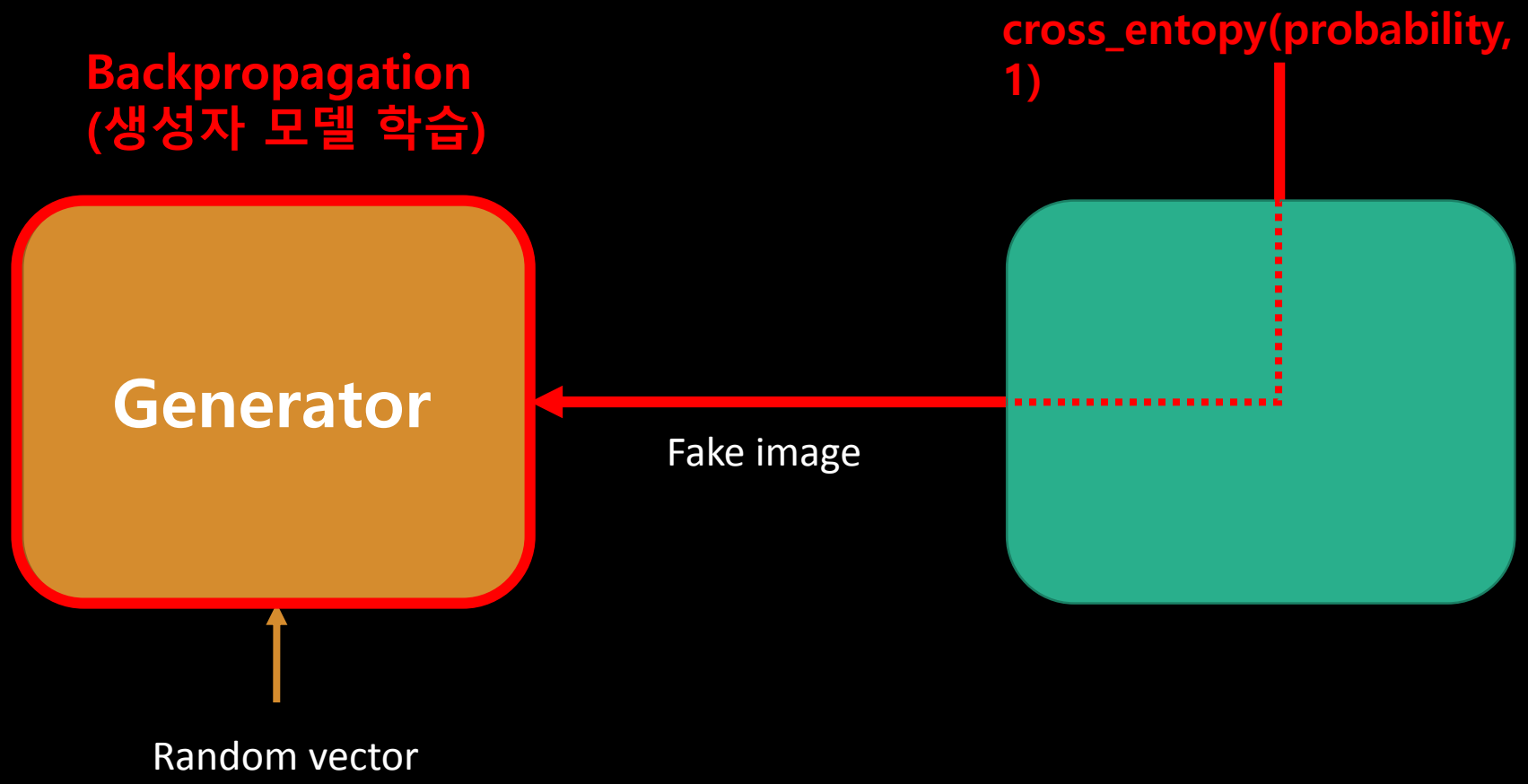
Generative Adversarial Network (GAN)



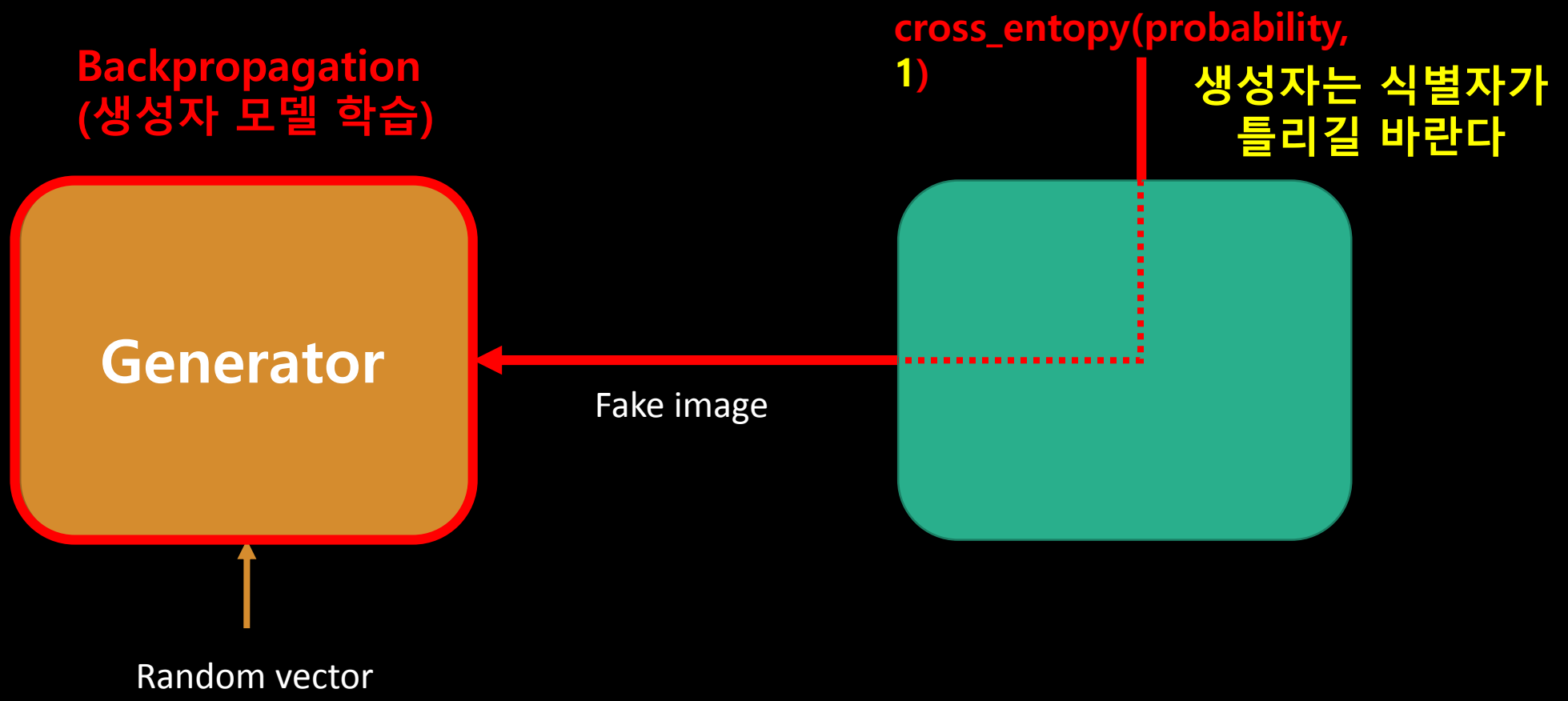
Generative Adversarial Network (GAN)



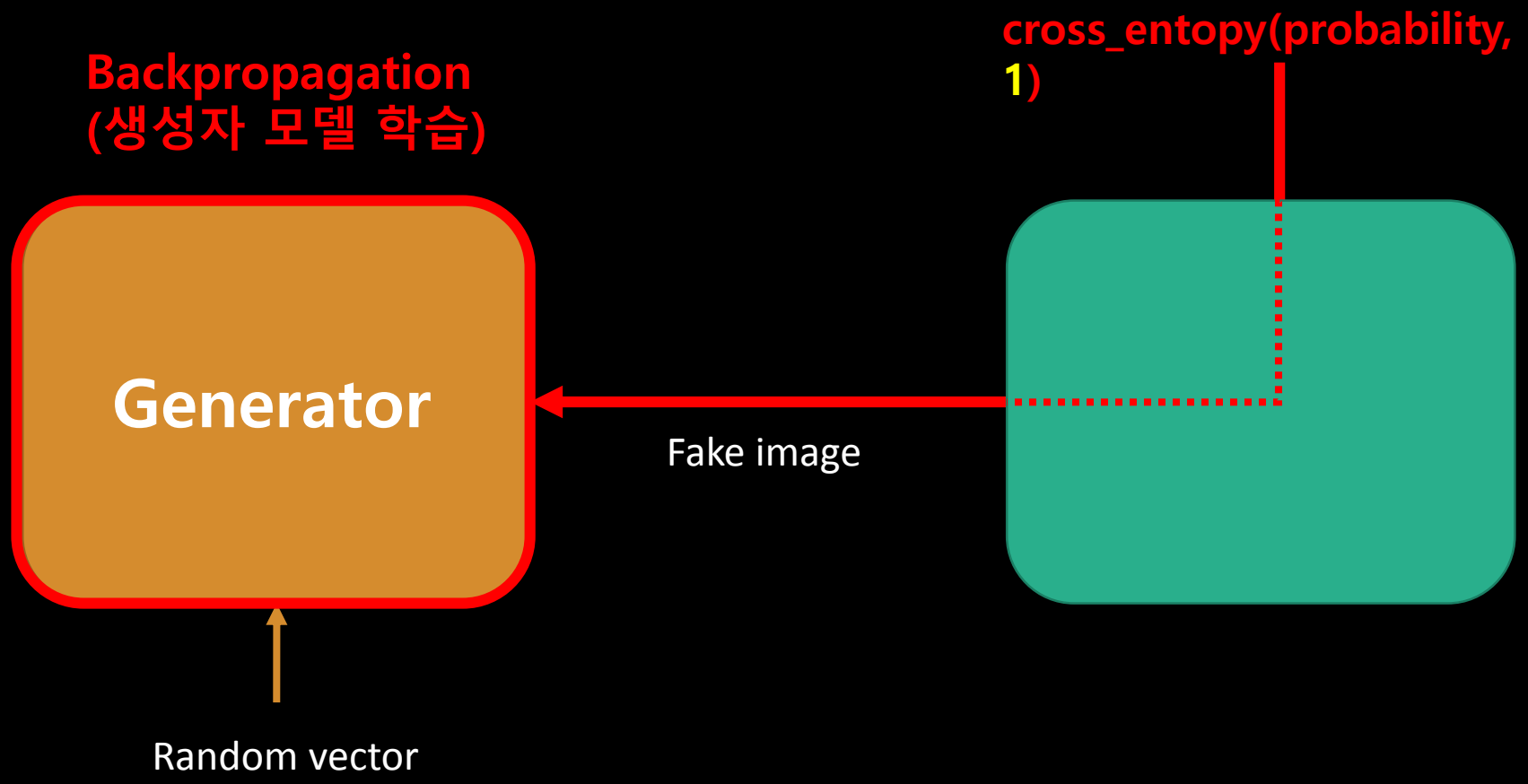
Generative Adversarial Network (GAN)



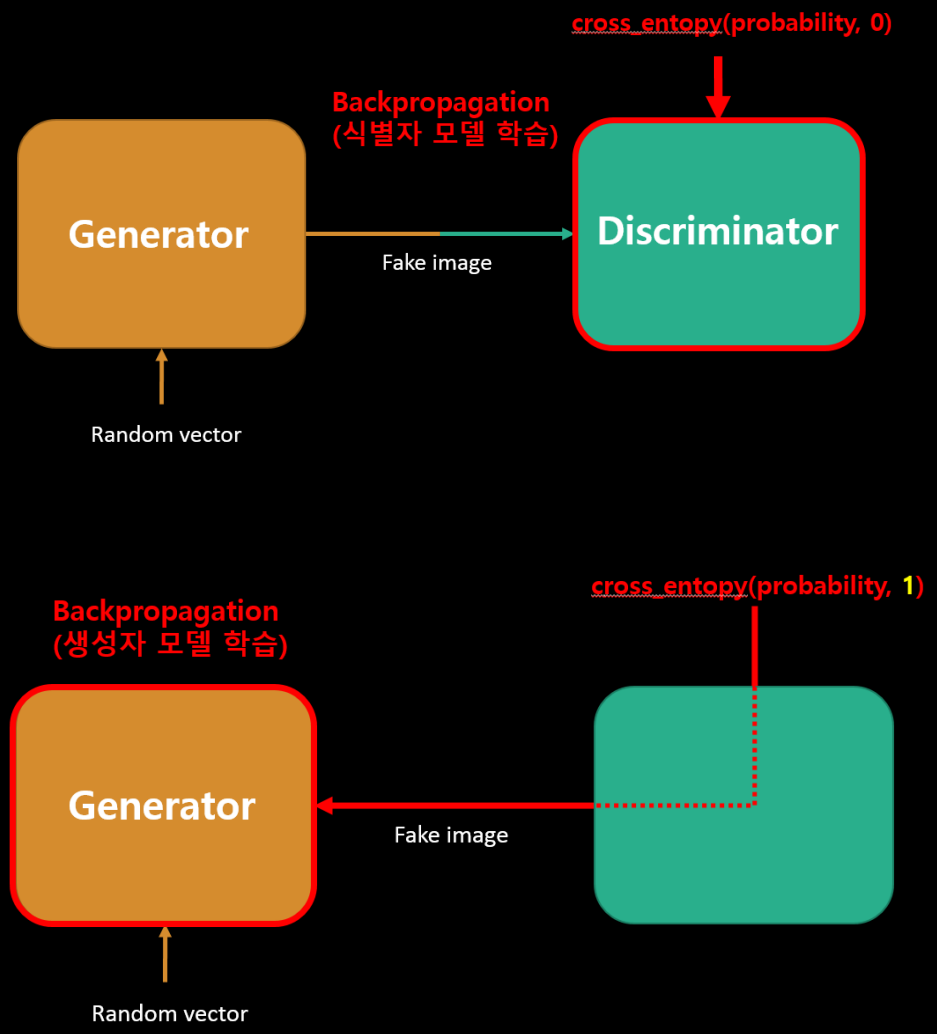
Generative Adversarial Network (GAN)



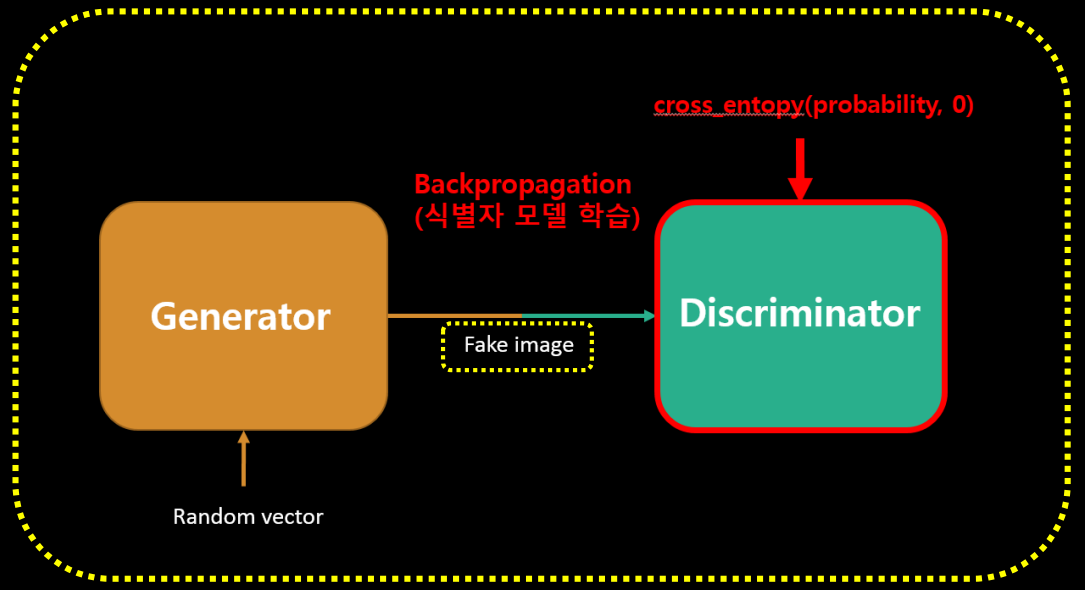
Generative Adversarial Network (GAN)



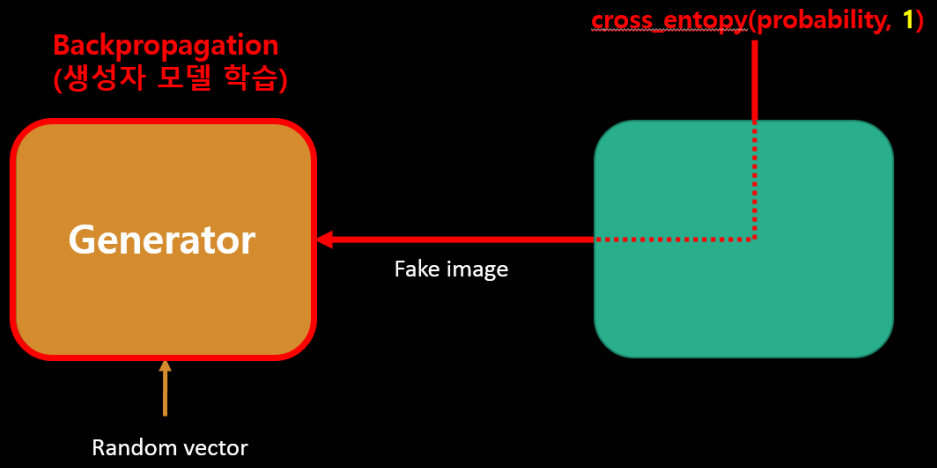
Generative Adversarial Network (GAN)



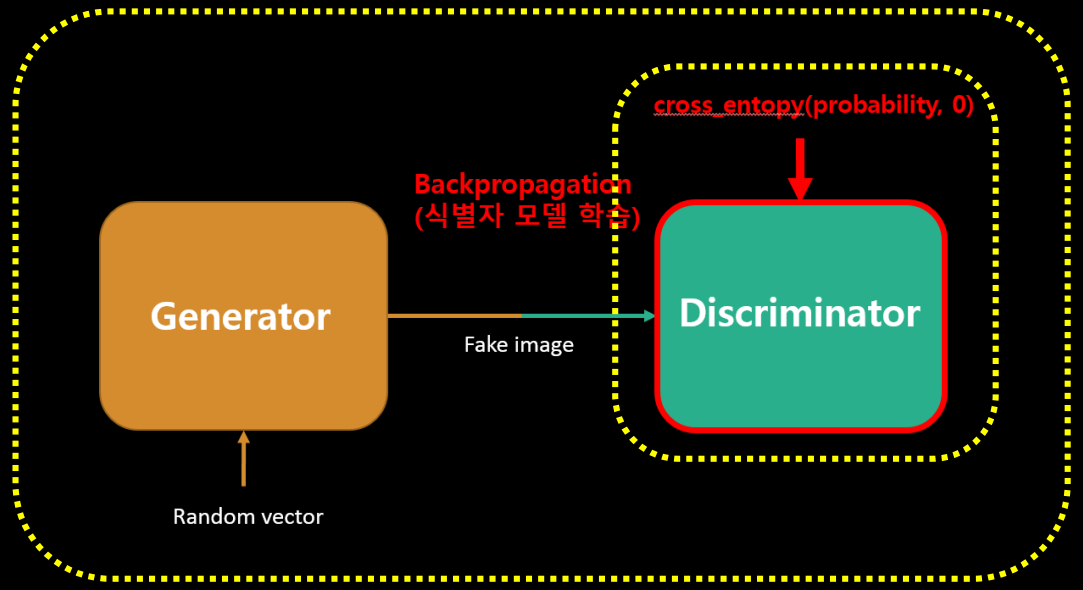
Generative Adversarial Network (GAN)



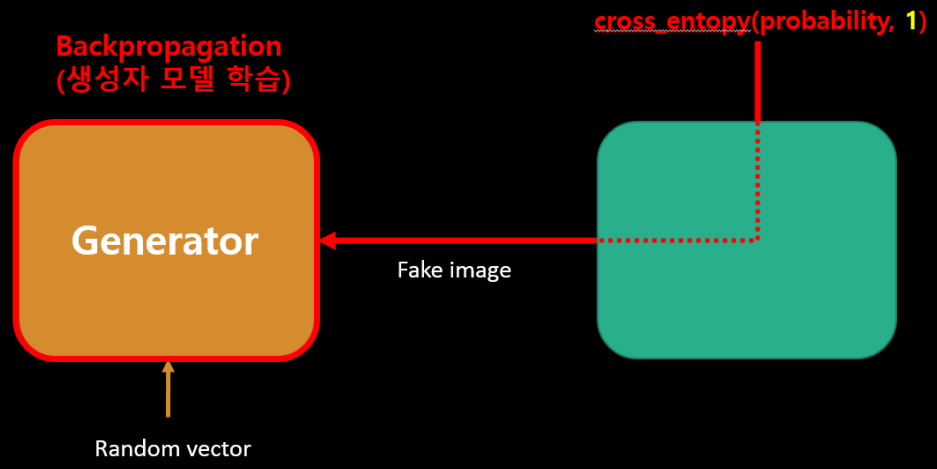
생성자는 가짜
이미지를 생성하고



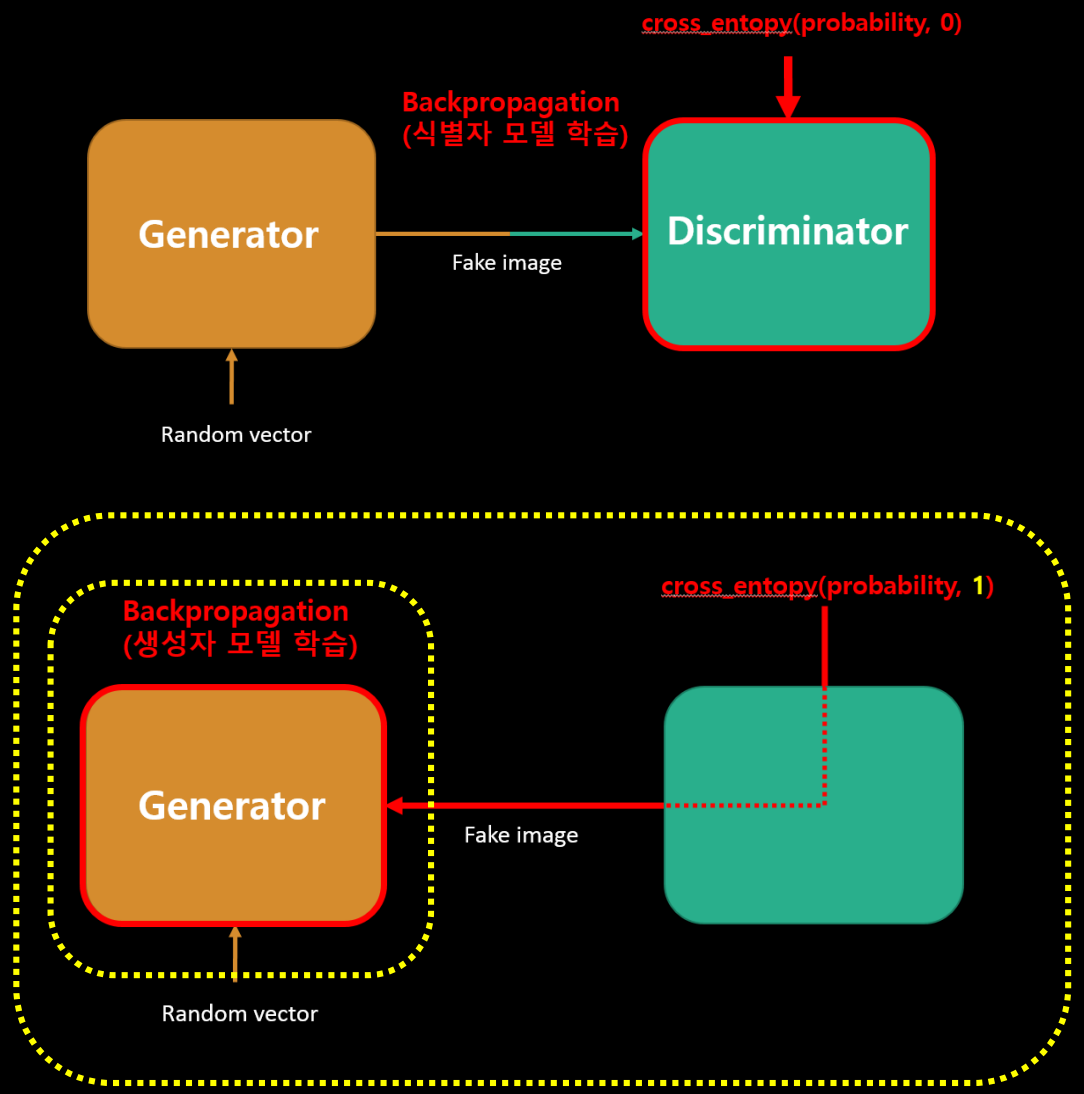
Generative Adversarial Network (GAN)



식별자는 가짜이미지를
'가짜'라고 판별하도록 학습

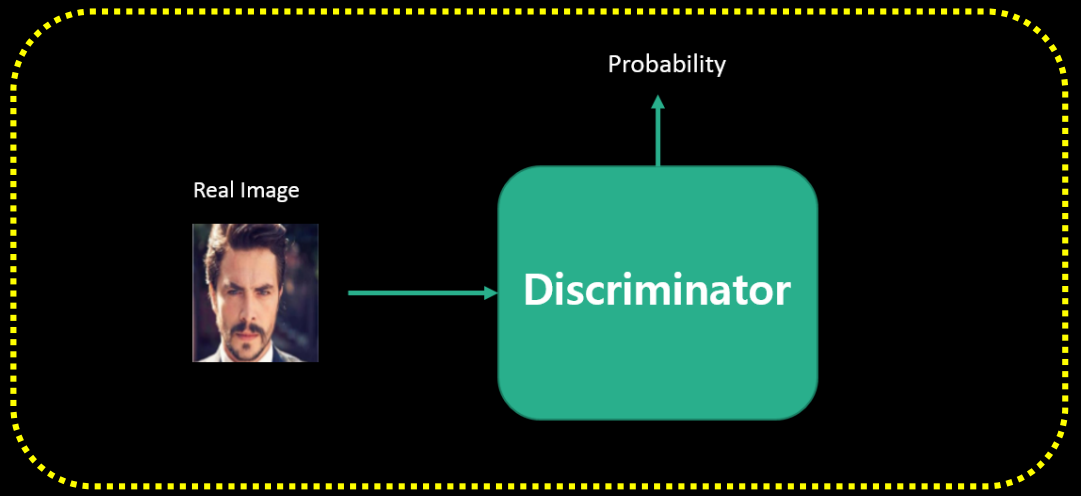


Generative Adversarial Network (GAN)

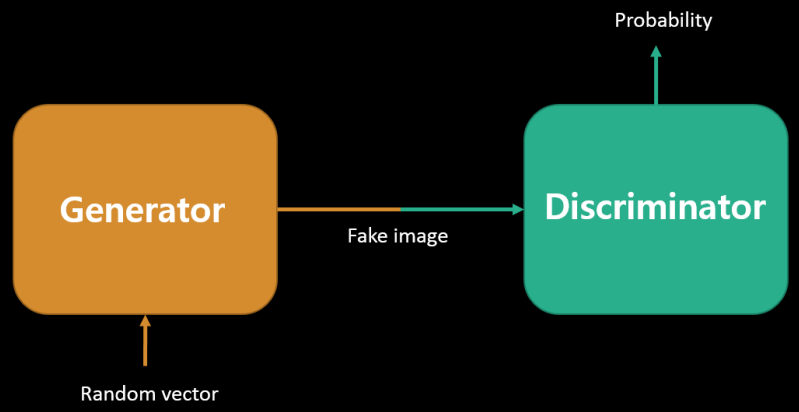


생성자는 식별자가 '가짜'를
'진짜'로 판별하도록 학습

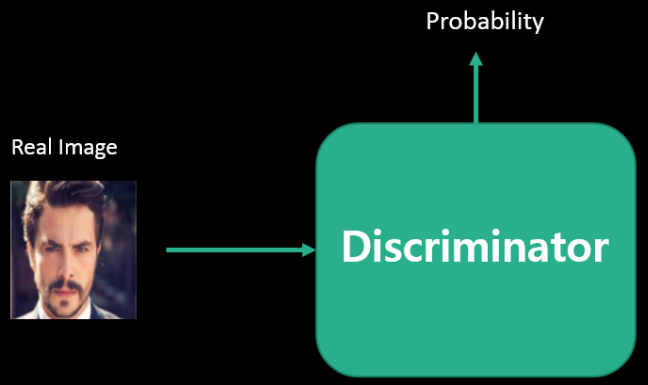
Generative Adversarial Network (GAN)

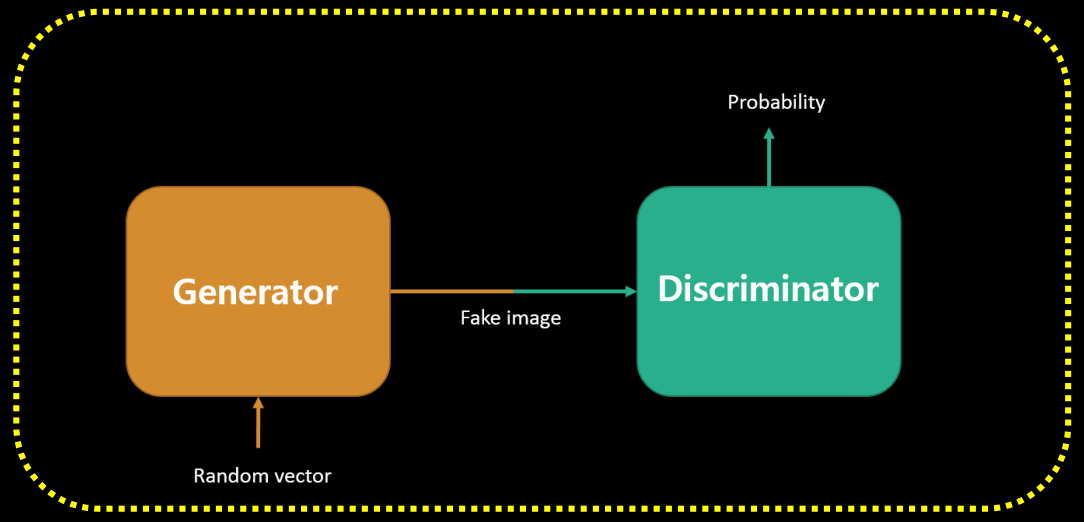


진짜 이미지를 가지고 학습할 때 구조
(식별자만 학습)



Generative Adversarial Network (GAN)

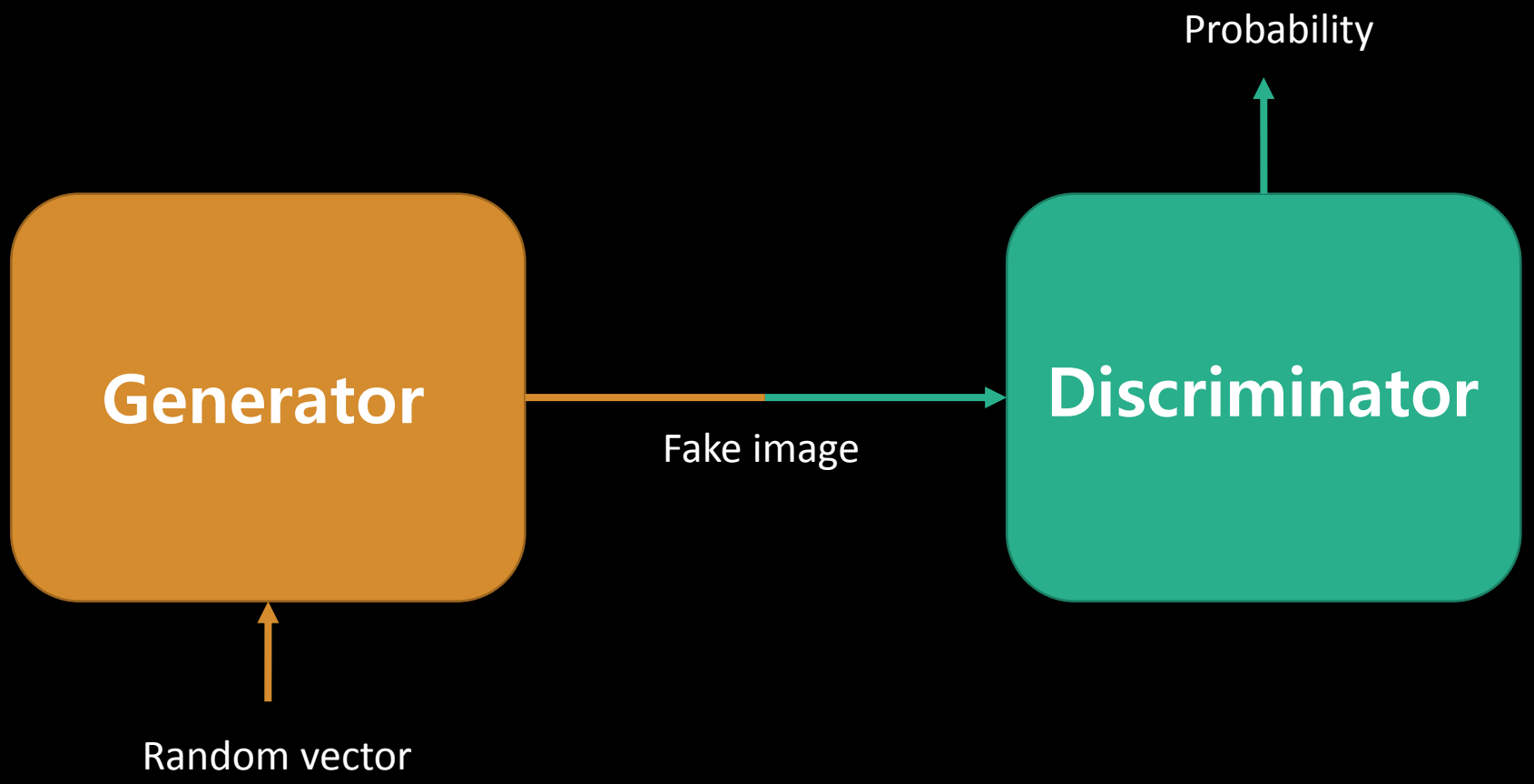




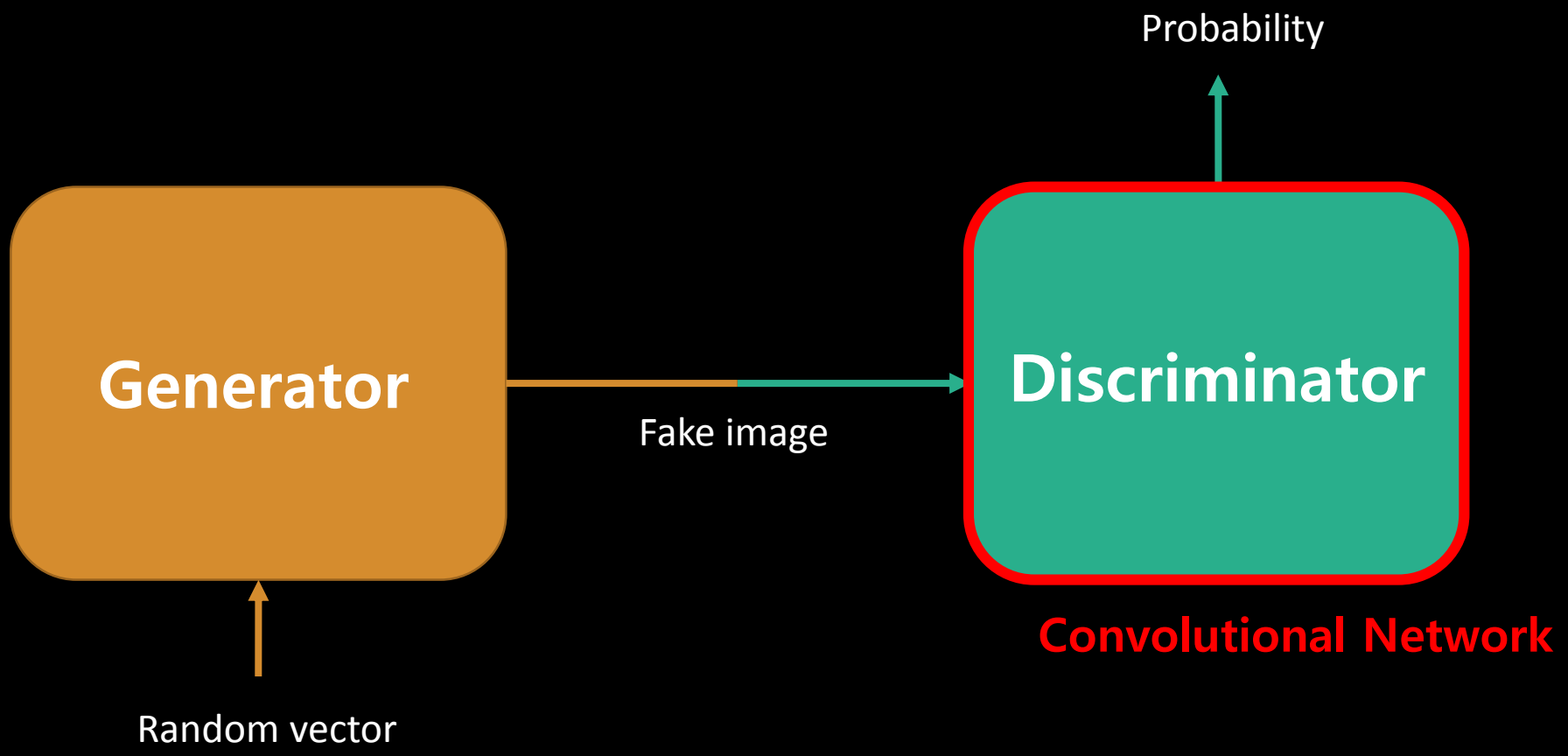
가짜 이미지를 가지고 학습할 때 구조
(식별자와 생성자 모두 학습)

2. DCGAN

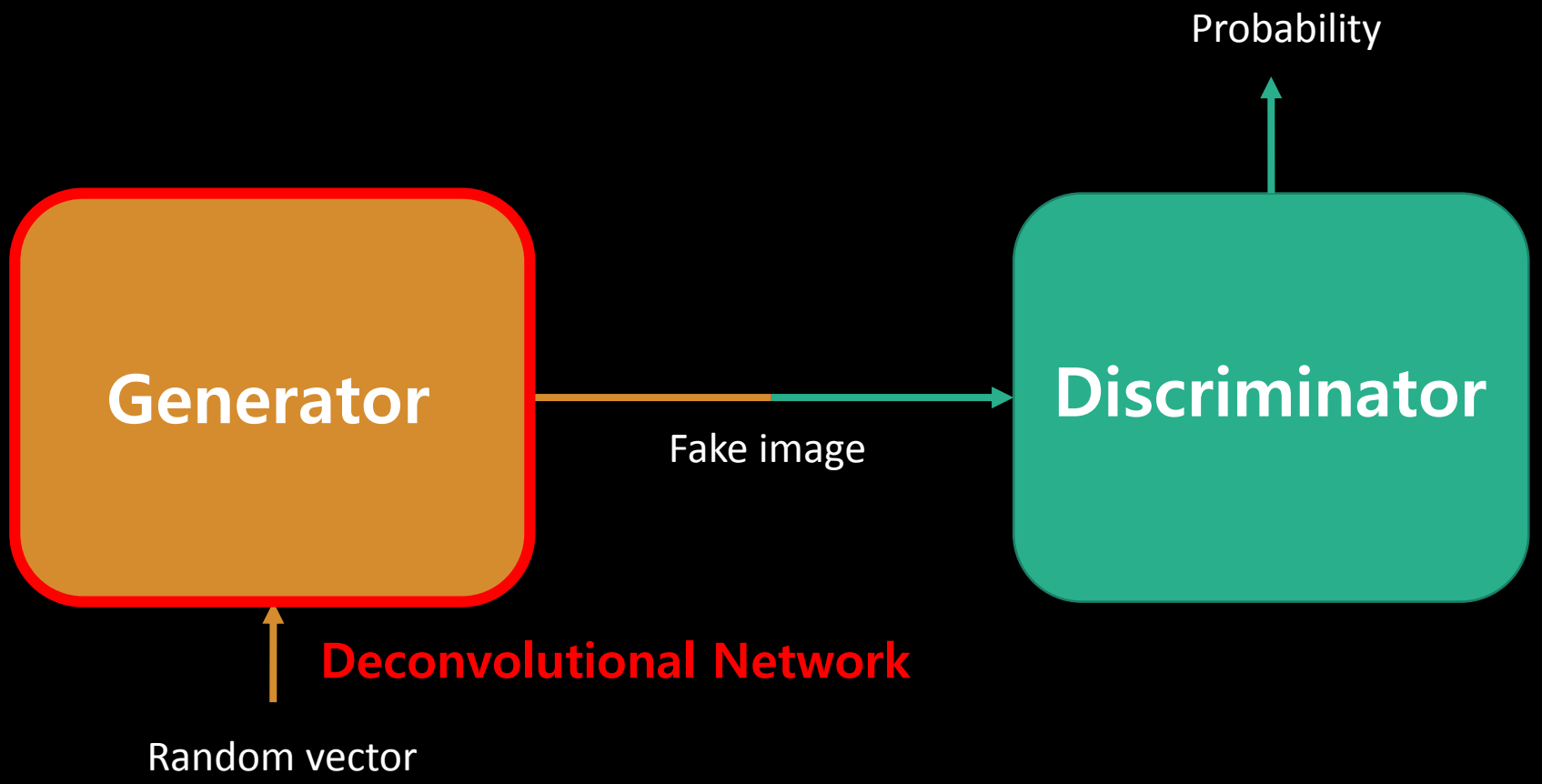
Deep Convolutional GAN (DCGAN)



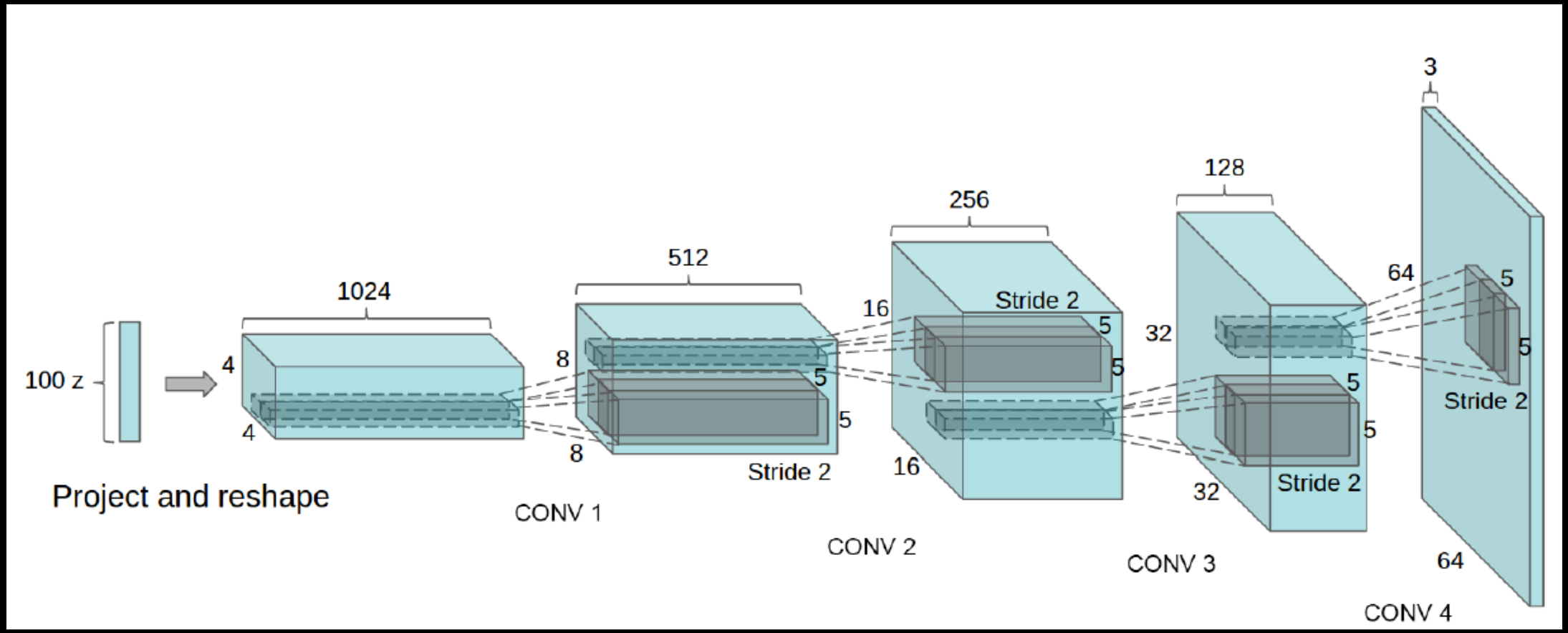
Deep Convolutional GAN (DCGAN)



Deep Convolutional GAN (DCGAN)



Deep Convolutional GAN (DCGAN)



Generator: Deconvolutional Neural Network

Deep Convolutional GAN (DCGAN)

Discriminator = Convolutional Network

Generator= Deconvolutional Network

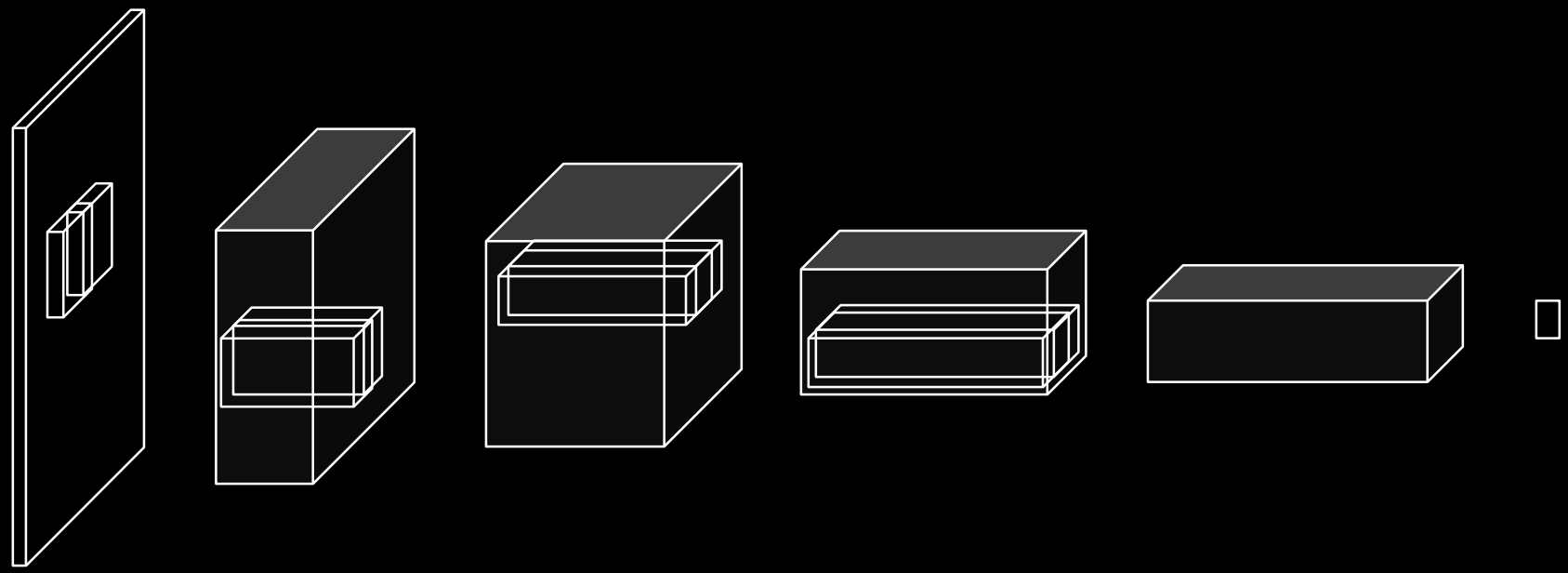
Deep Convolutional GAN (DCGAN)

DCGAN 모델의 특징

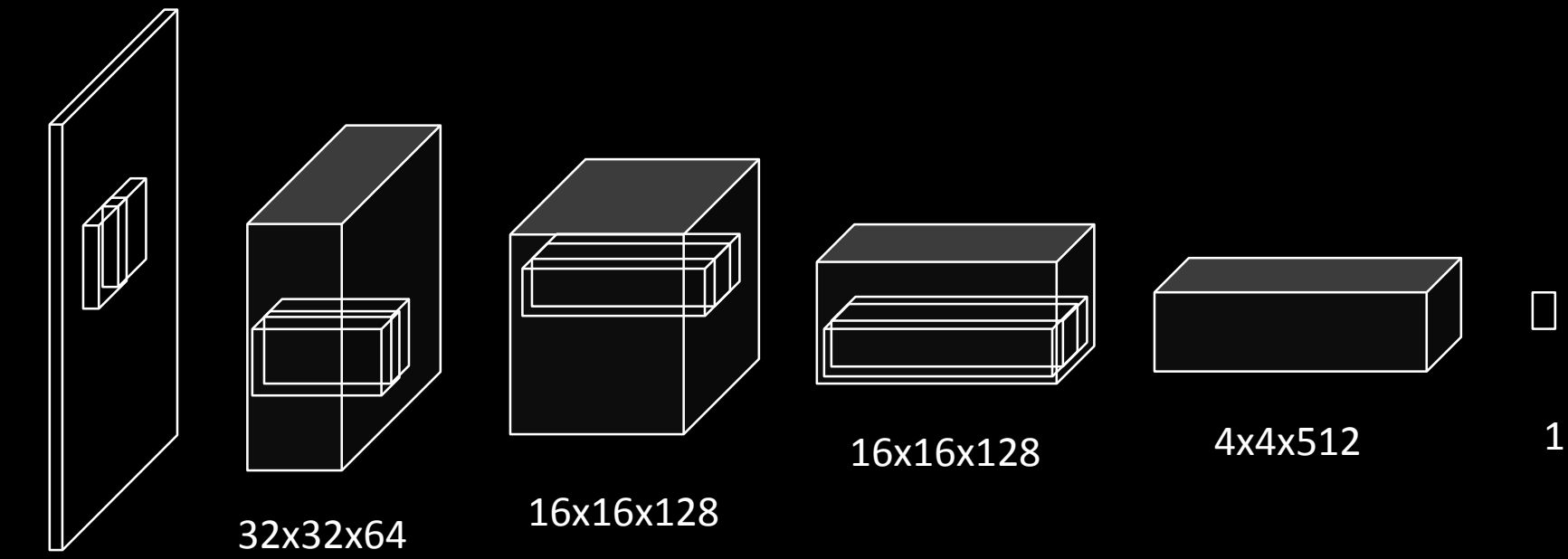
1. Pooling Layer를 사용하지 않음
2. Batch Normalization 사용
3. Fully Connected Layer 최소화
4. ReLU와 Leaky ReLU 사용

결과적으로 GAN의 학습을 안정화시켜 퀄리티가 더 높은 이미지를 생성

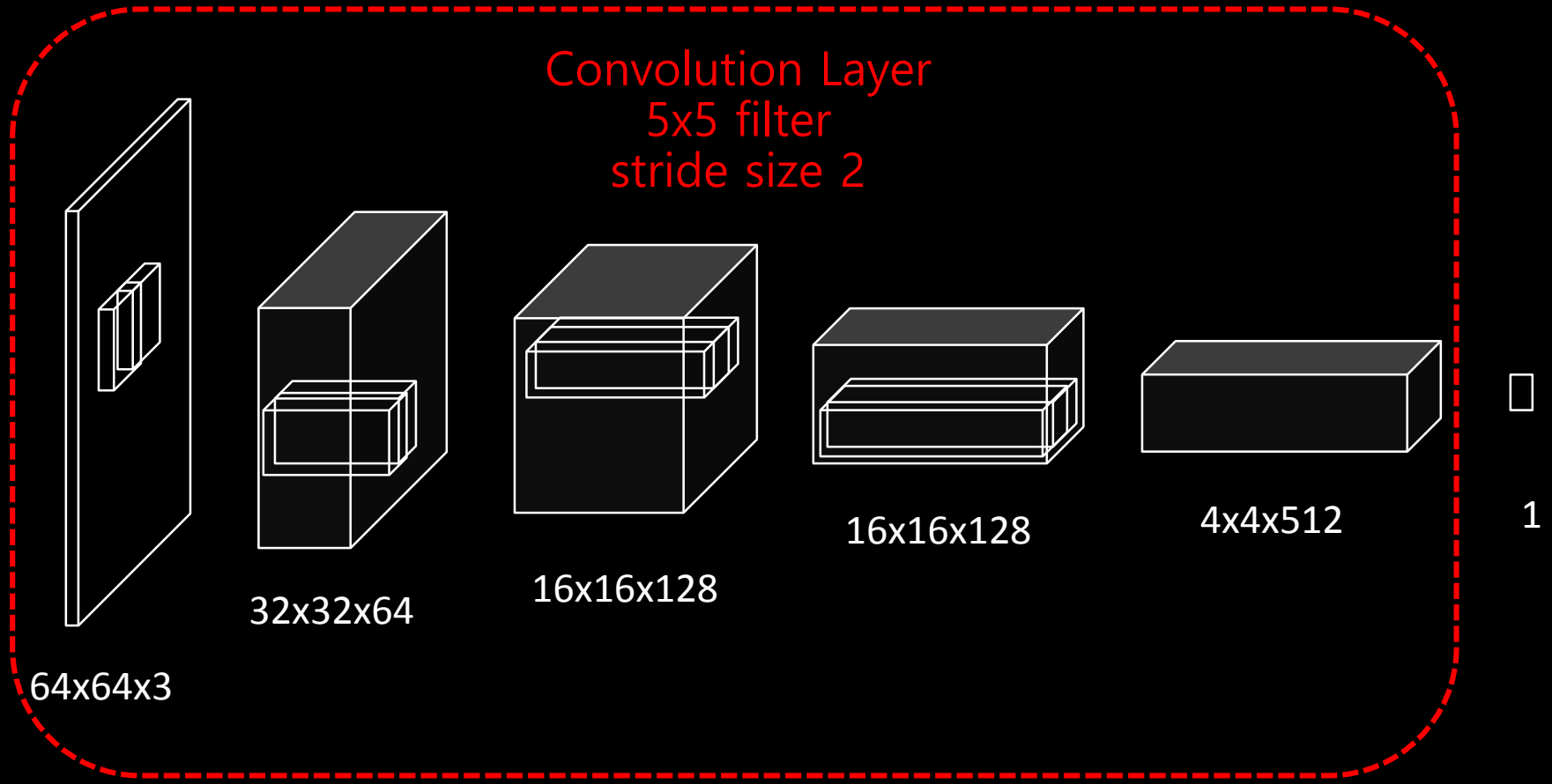
Discriminator(식별자)



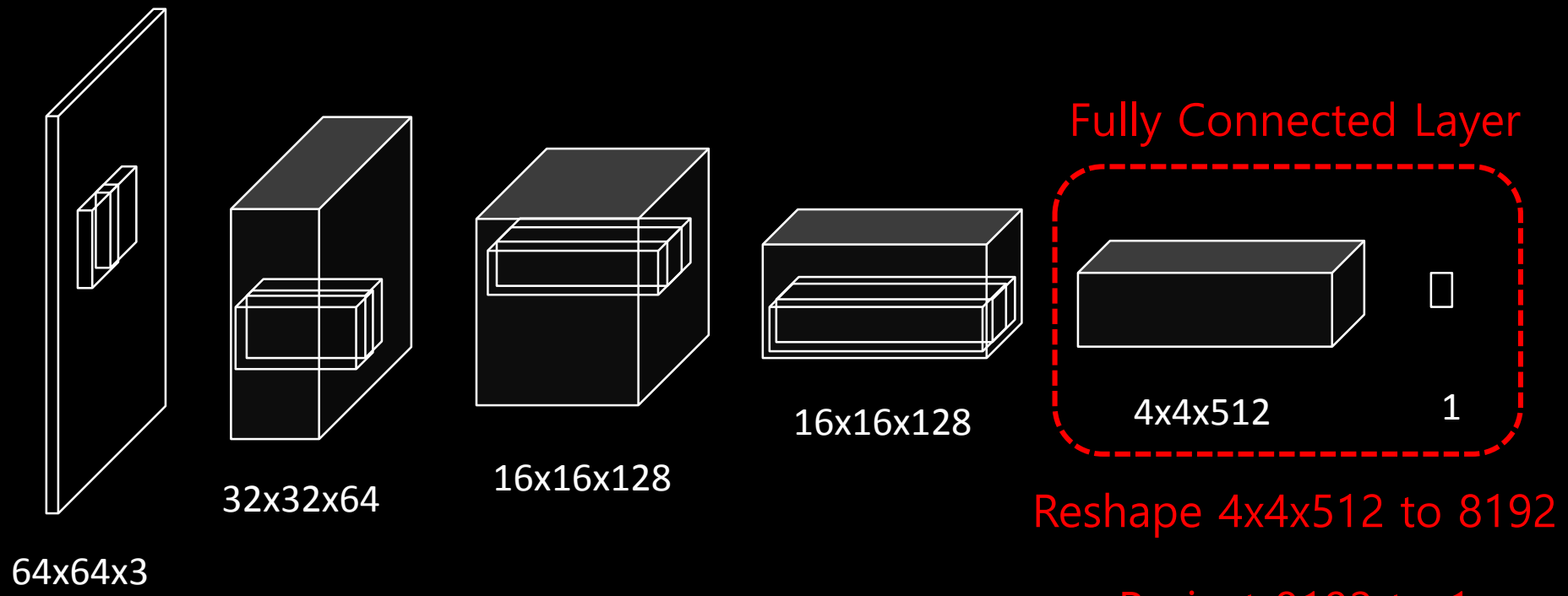
Discriminator(식별자)



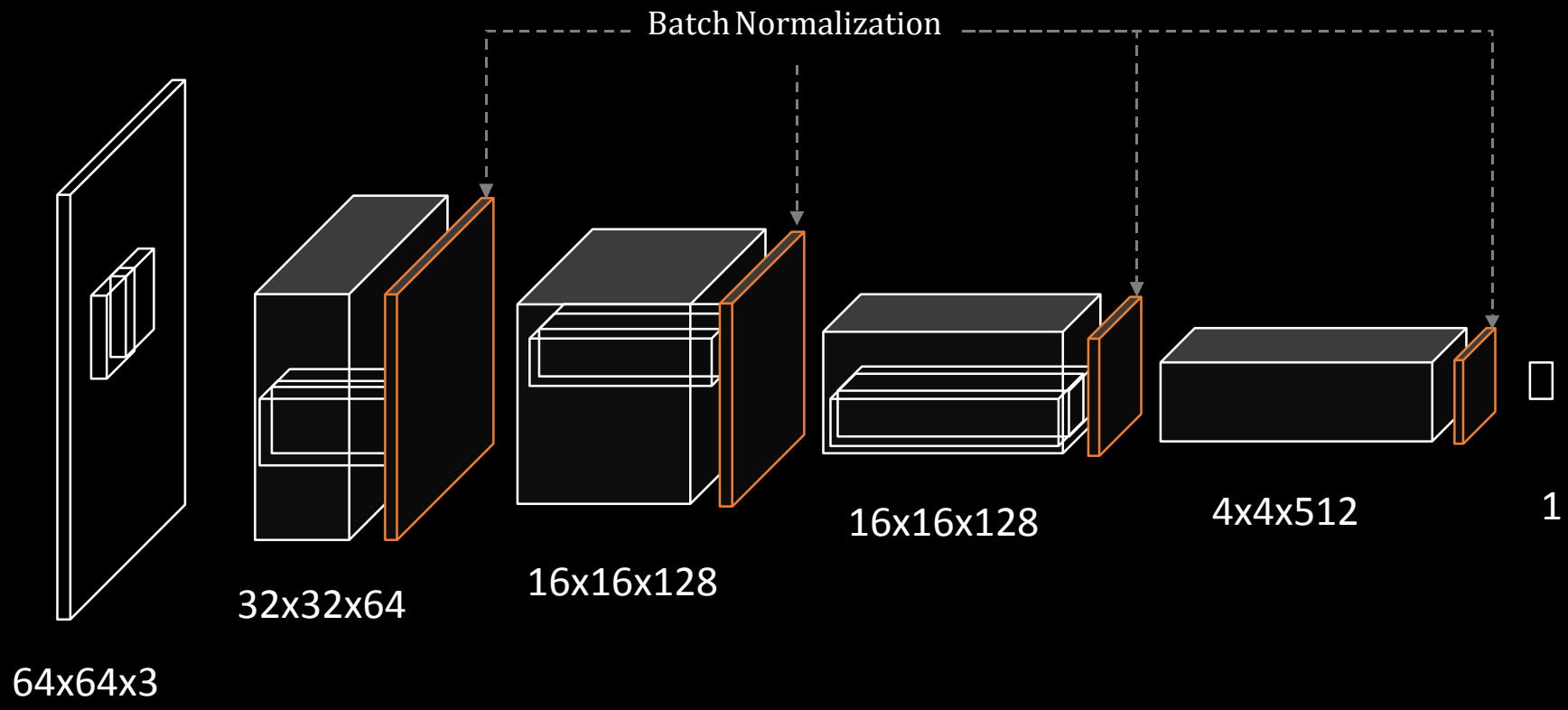
Discriminator(식별자)



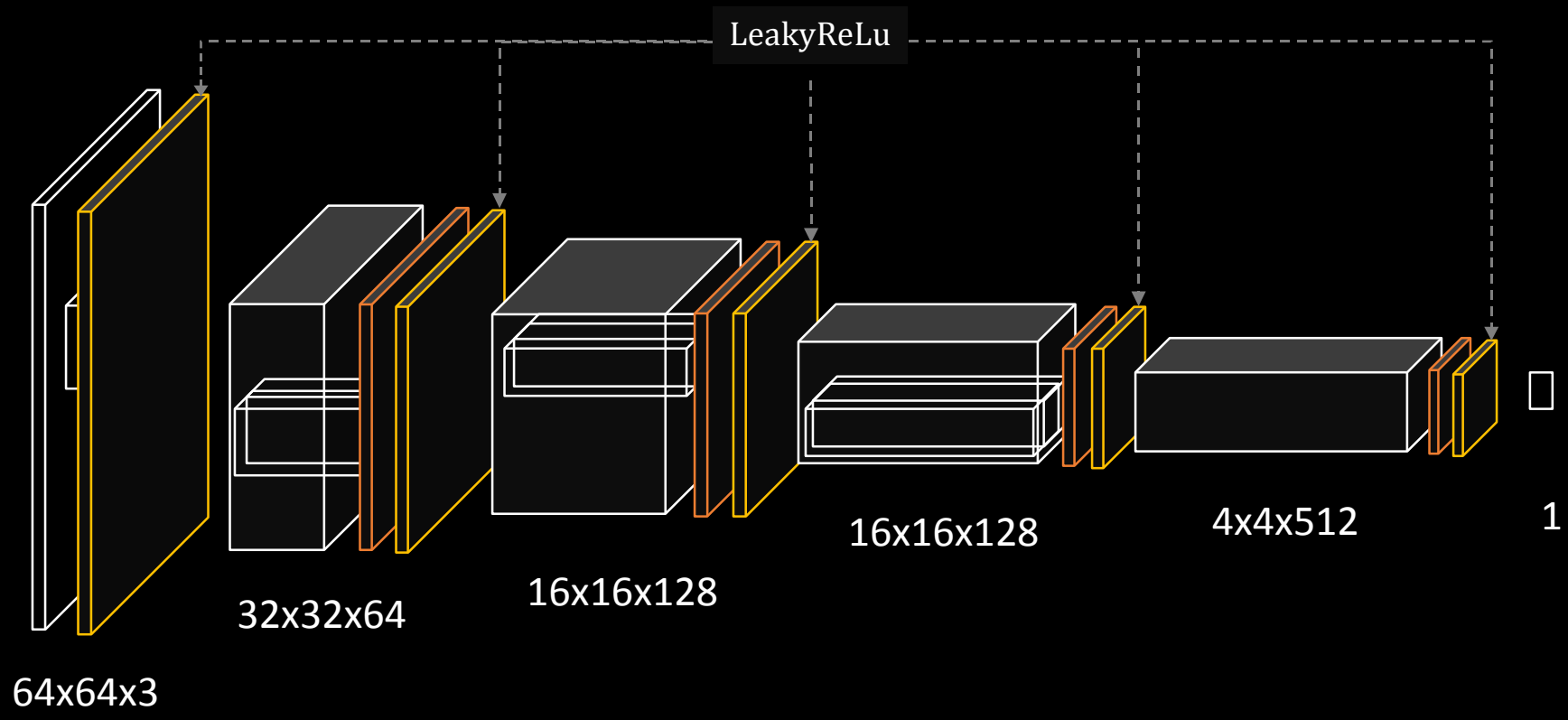
Discriminator(식별자)



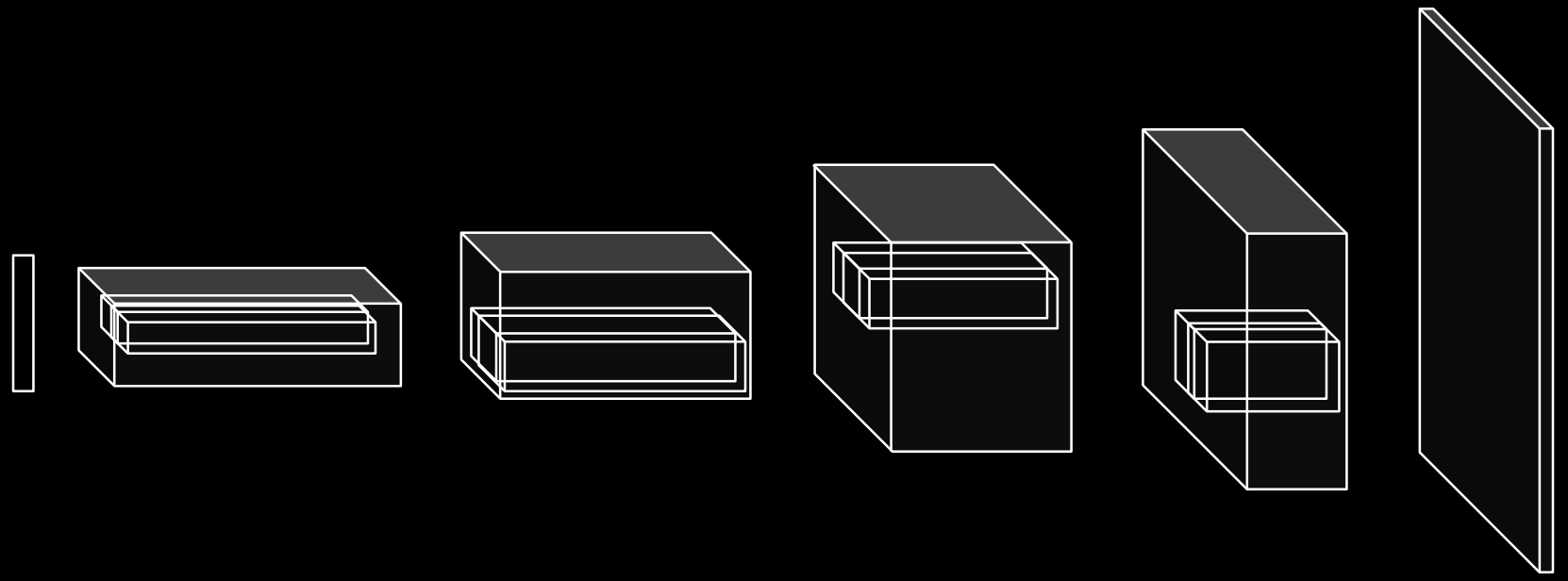
Discriminator(식별자)



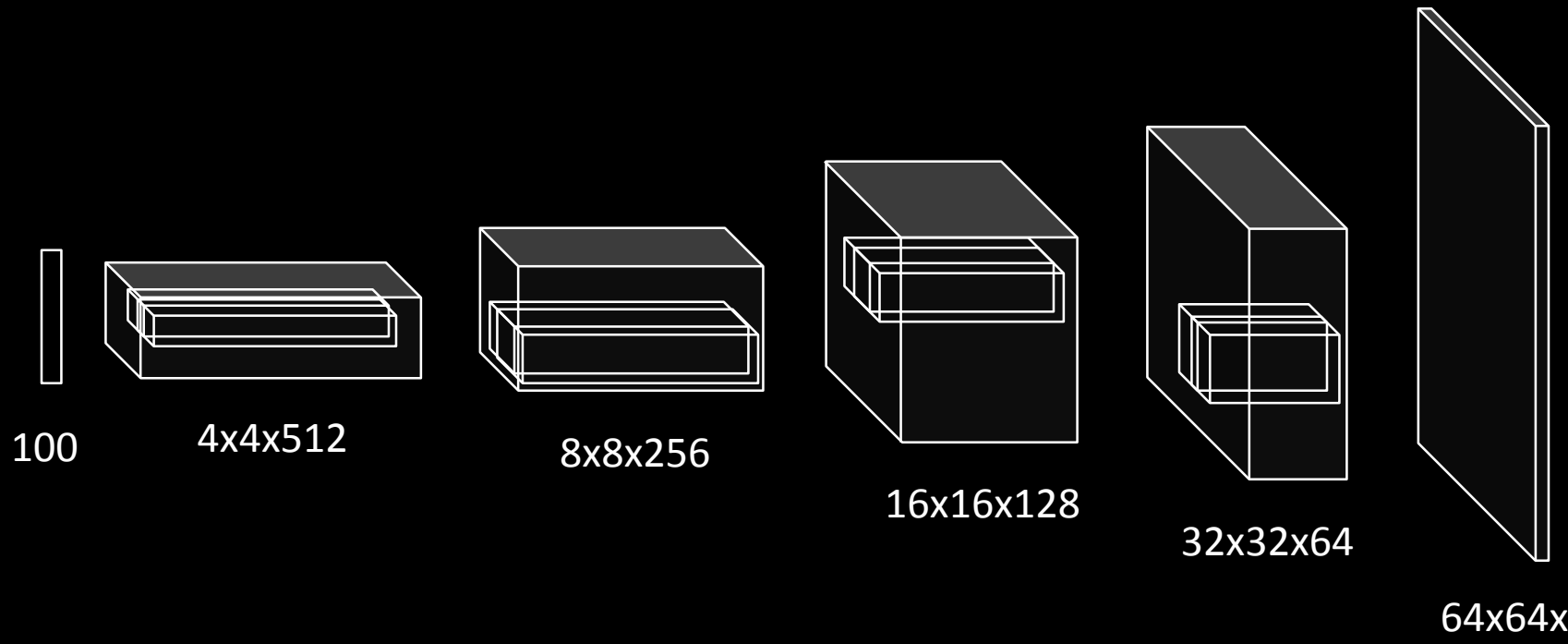
Discriminator(식별자)



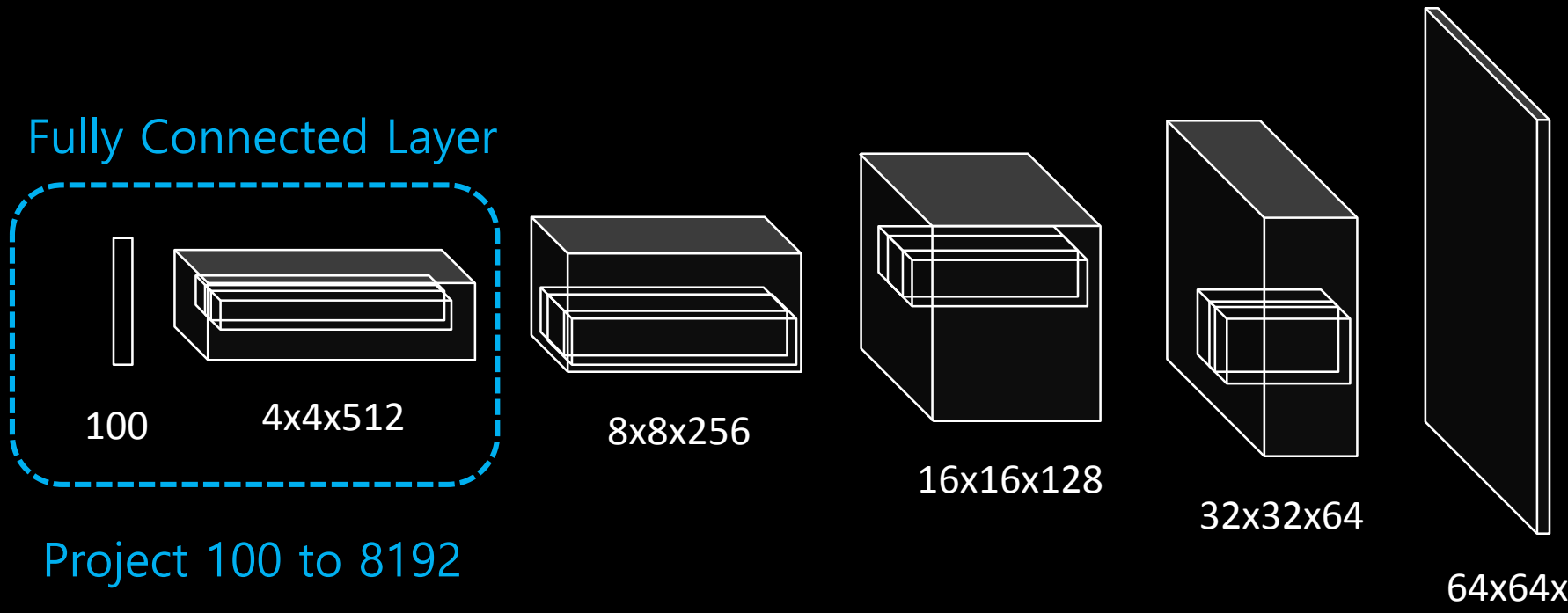
Generator(생성자)



Generator(생성자)



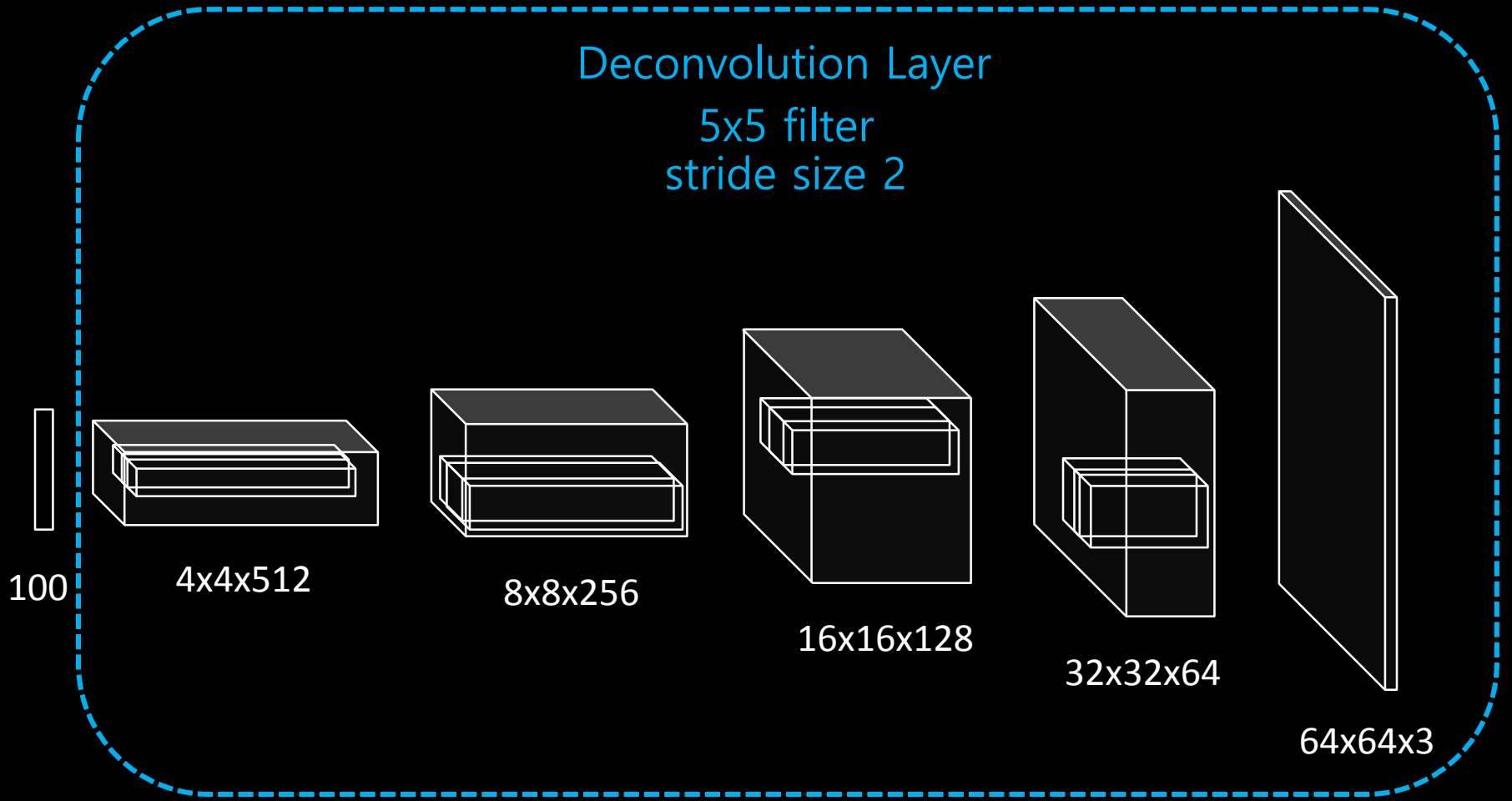
Generator(생성자)



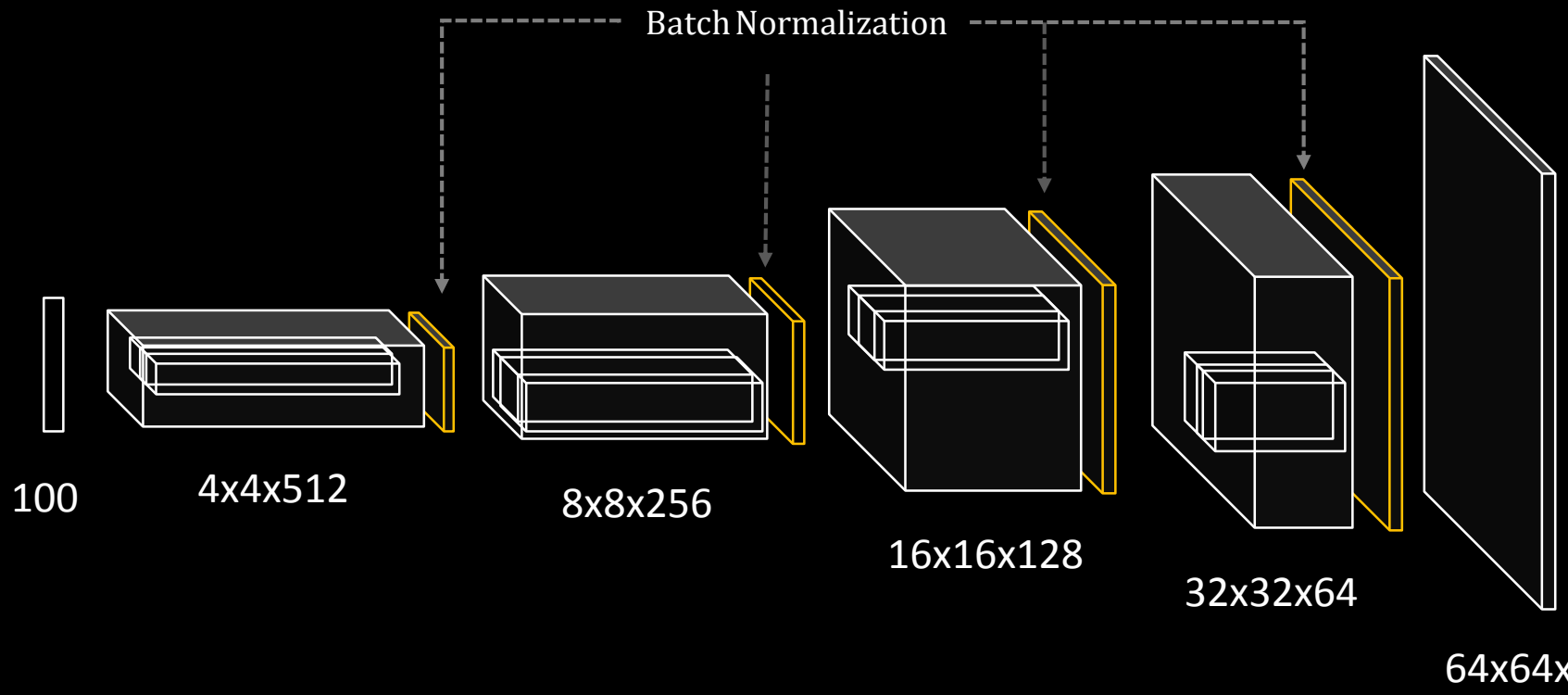
Project 100 to 8192

Reshape 8192 to 4x4x512

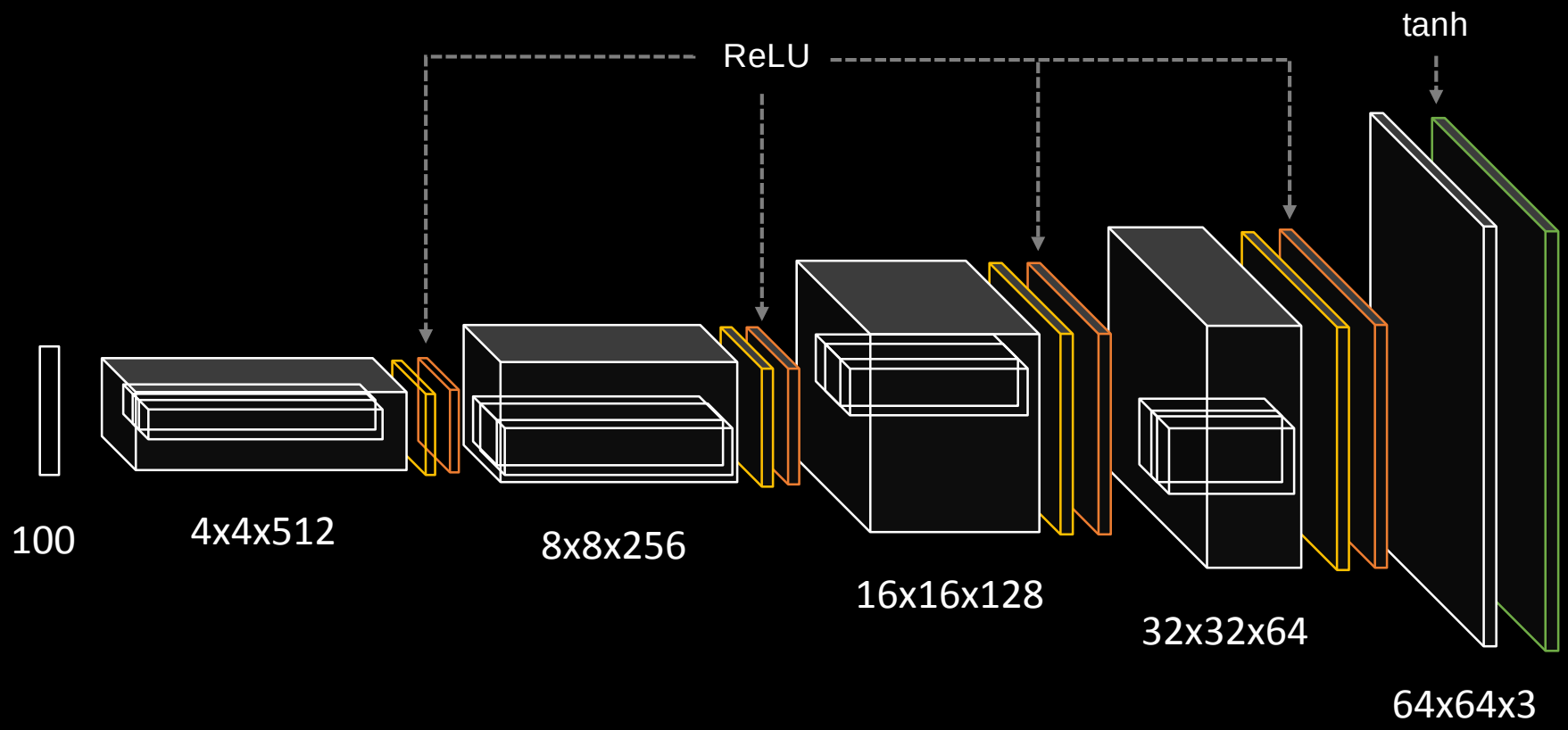
Generator(생성자)



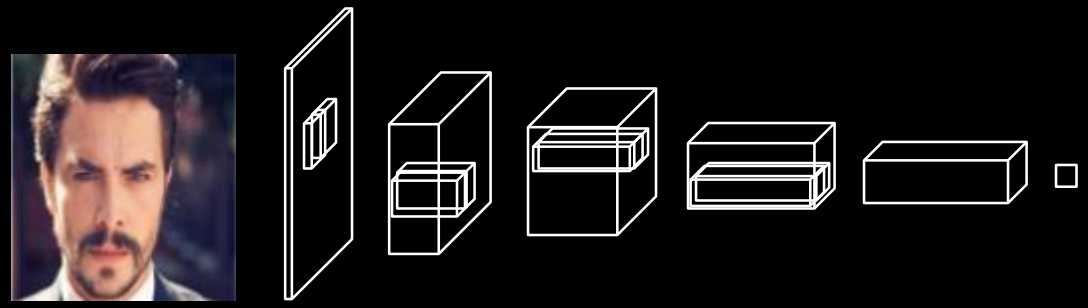
Generator(생성자)



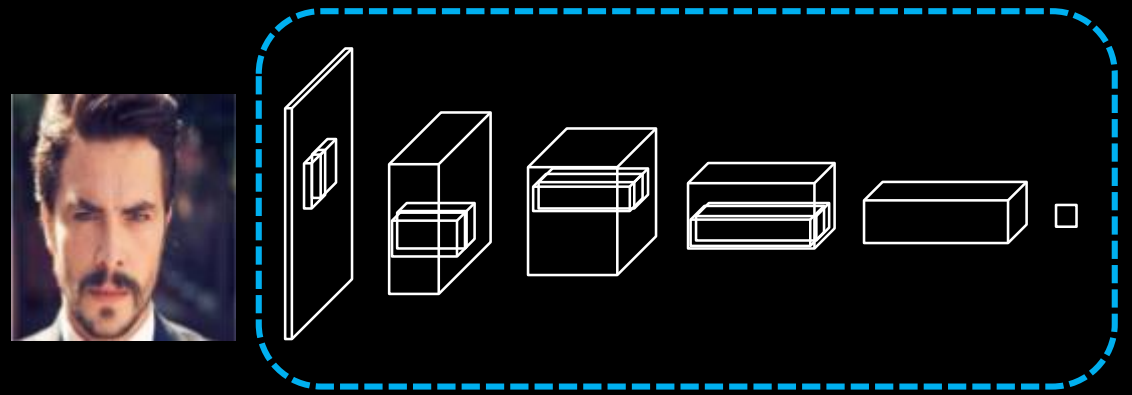
Generator(생성자)



진짜 이미지

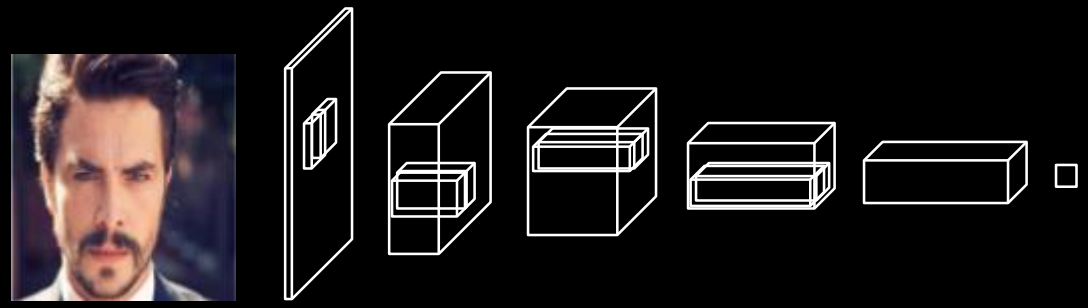


진짜 이미지

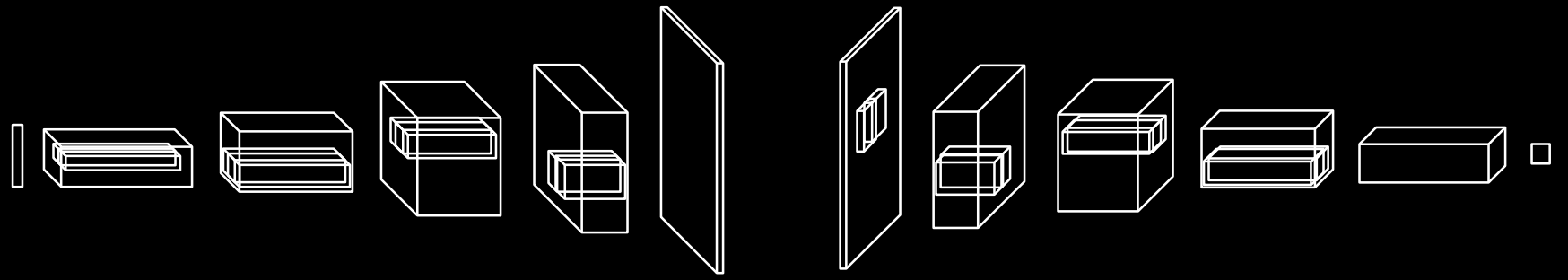


1이 나오도록
식별자만 학습

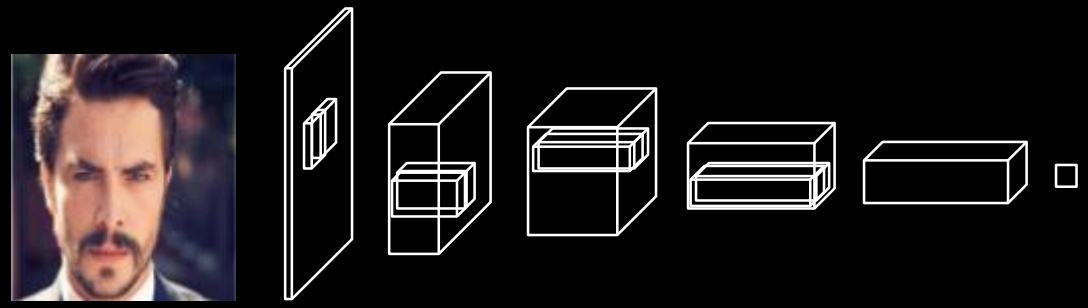
진짜 이미지



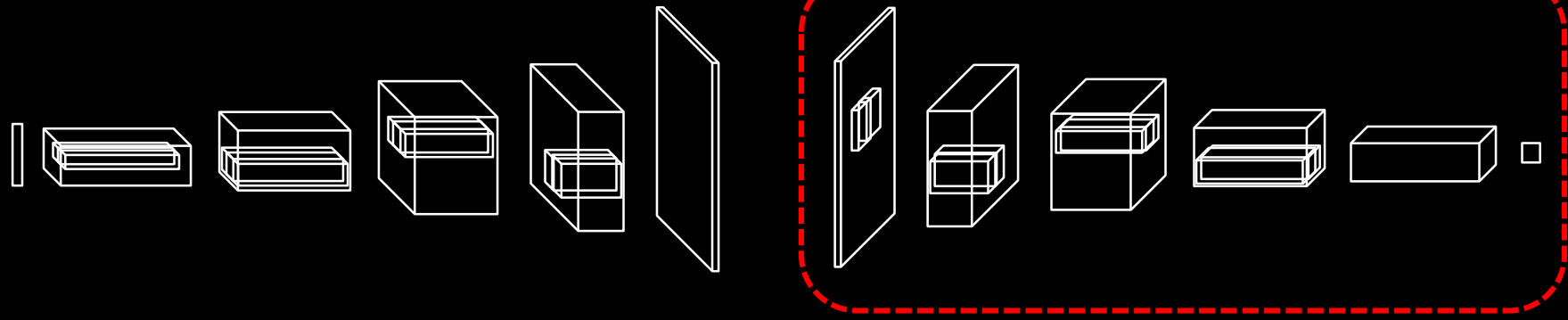
가짜 이미지



진짜 이미지

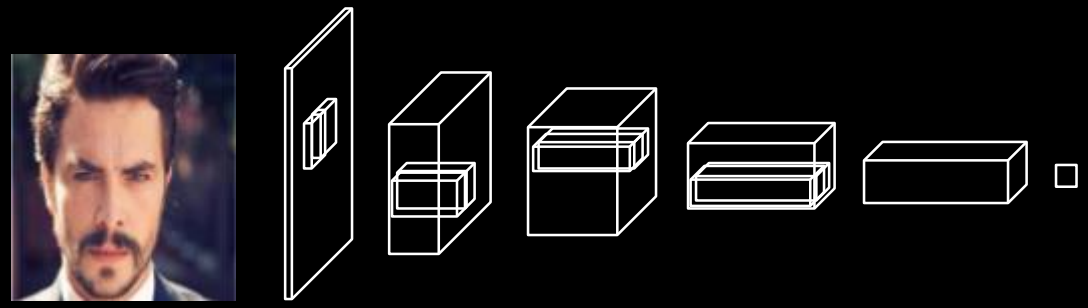


가짜 이미지

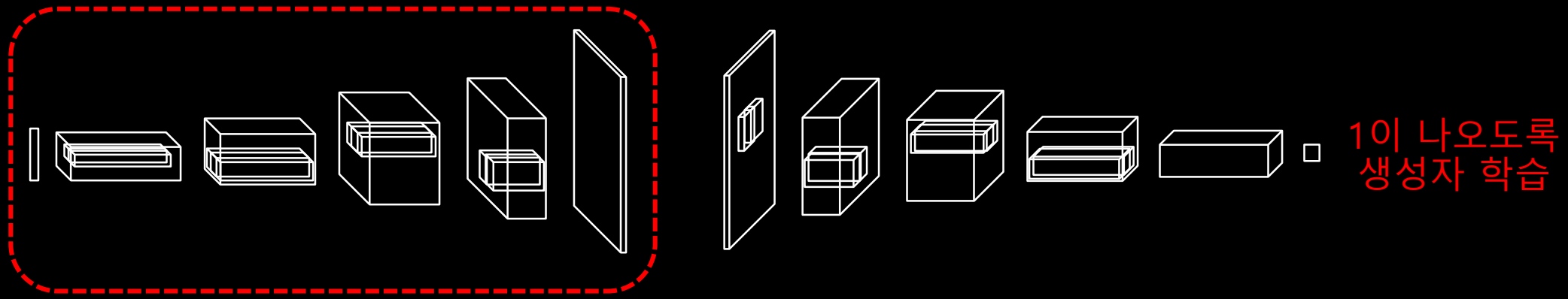


0이 나오도록
식별자 학습

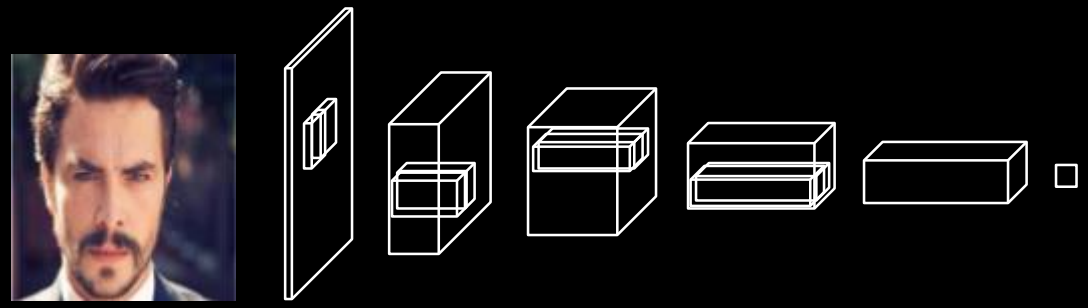
진짜 이미지



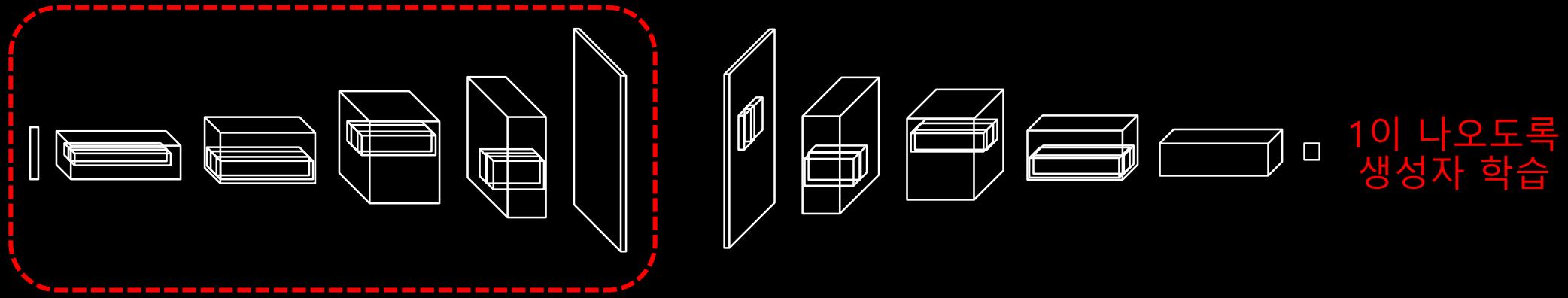
가짜 이미지



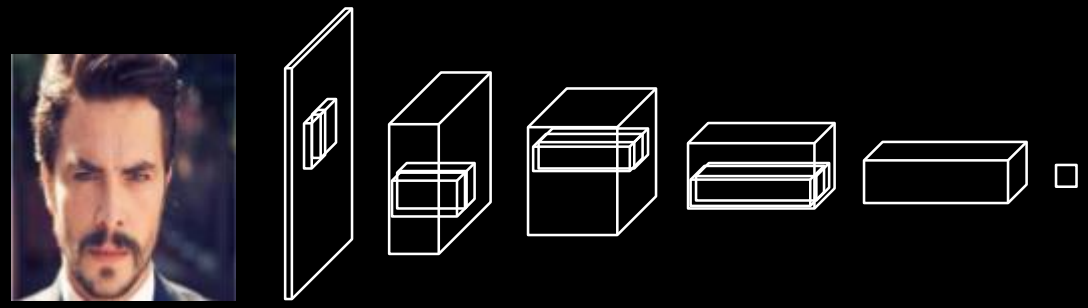
진짜 이미지



가짜 이미지

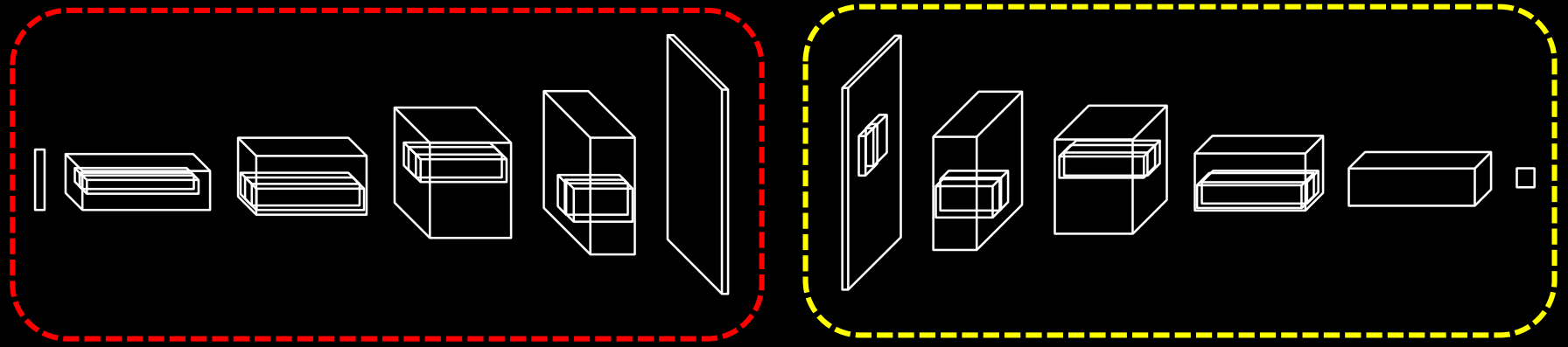


진짜 이미지



가짜 이미지

이쪽은 backprop만 되고
weight 학습 x



1이 나오도록
생성자 학습

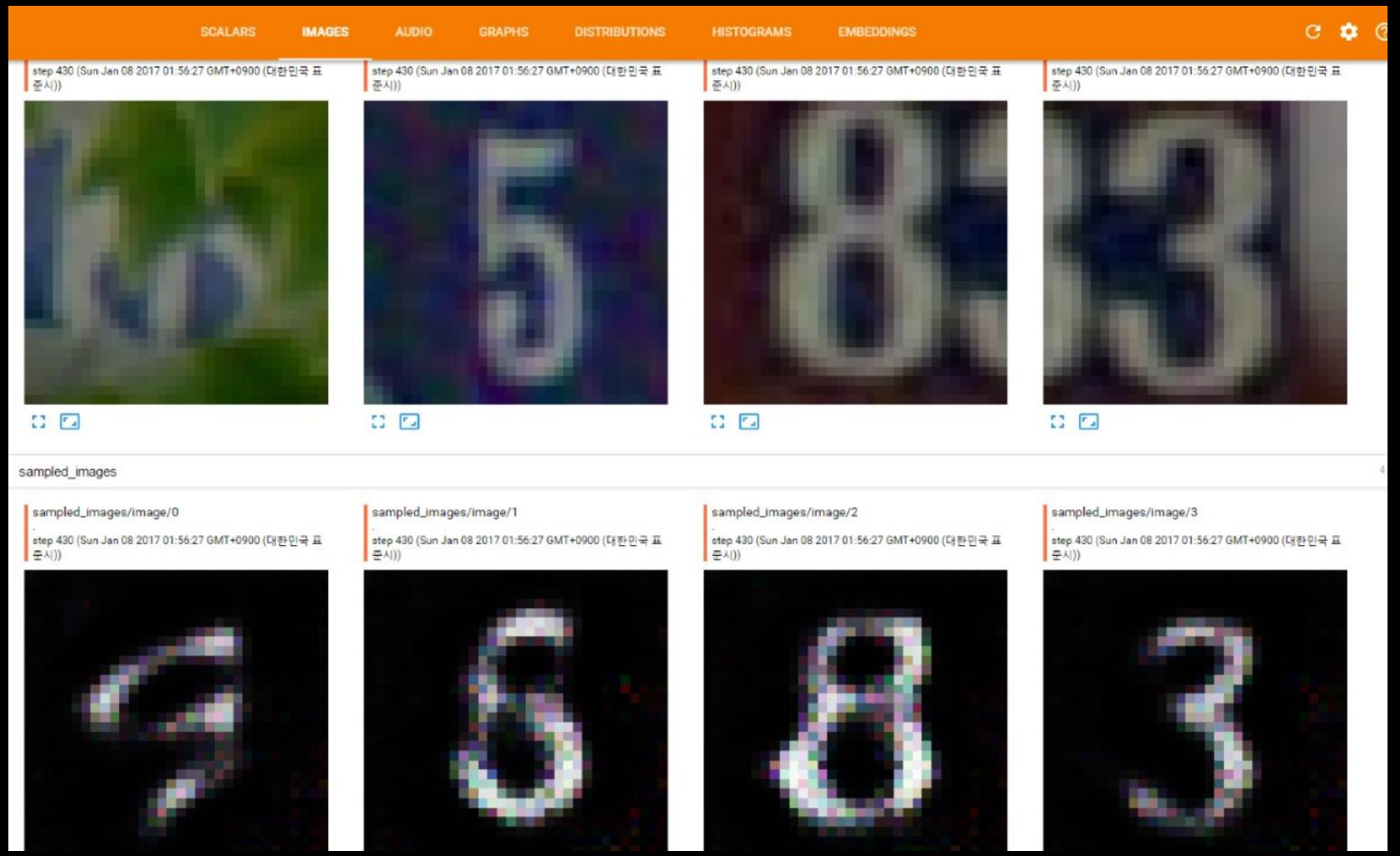
backpropagation

3. Domain Transfer Network

Domain Transfer Network (DTN)

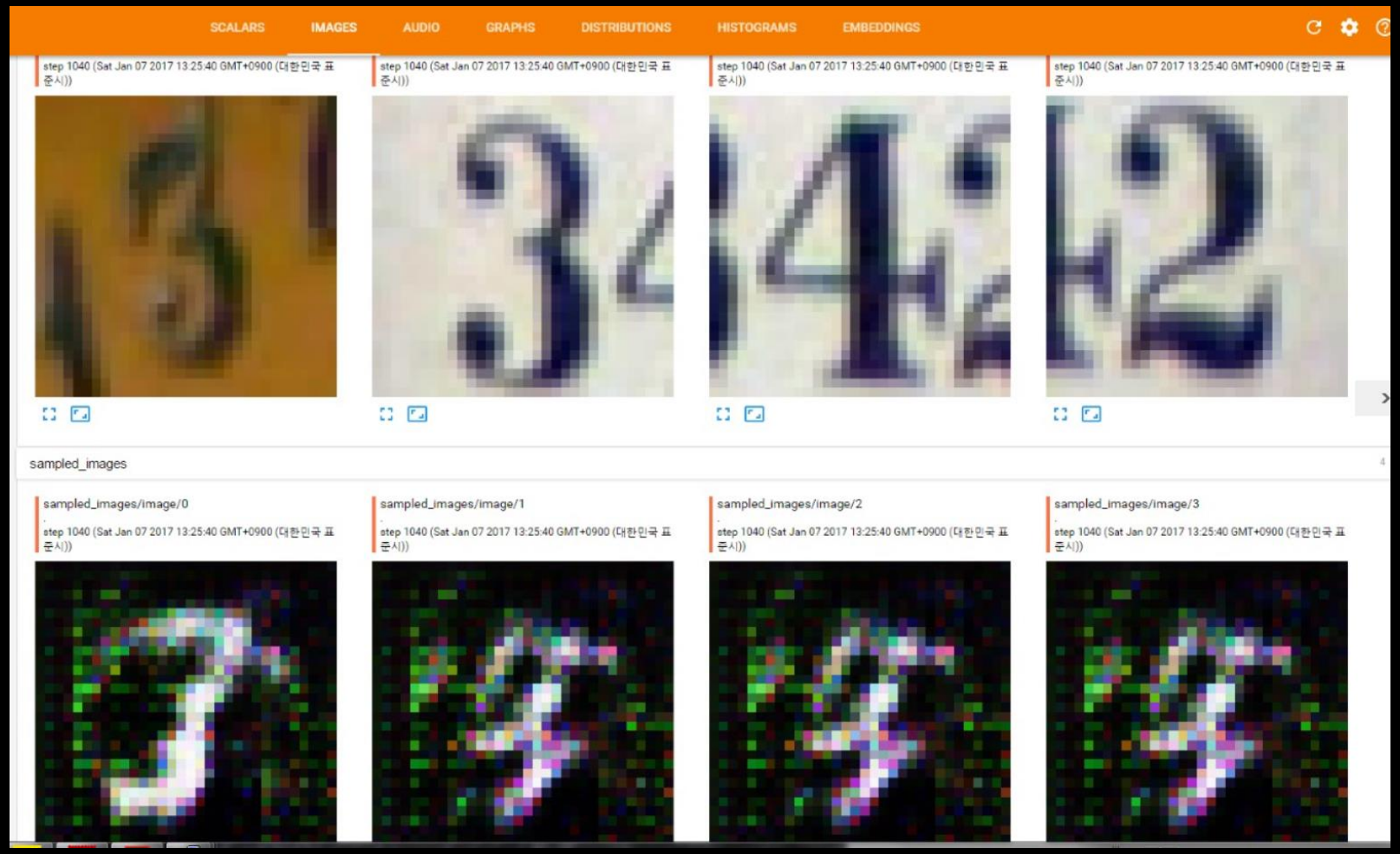
Transfer from SVHN to MNIST

Domain Transfer Network (DTN)



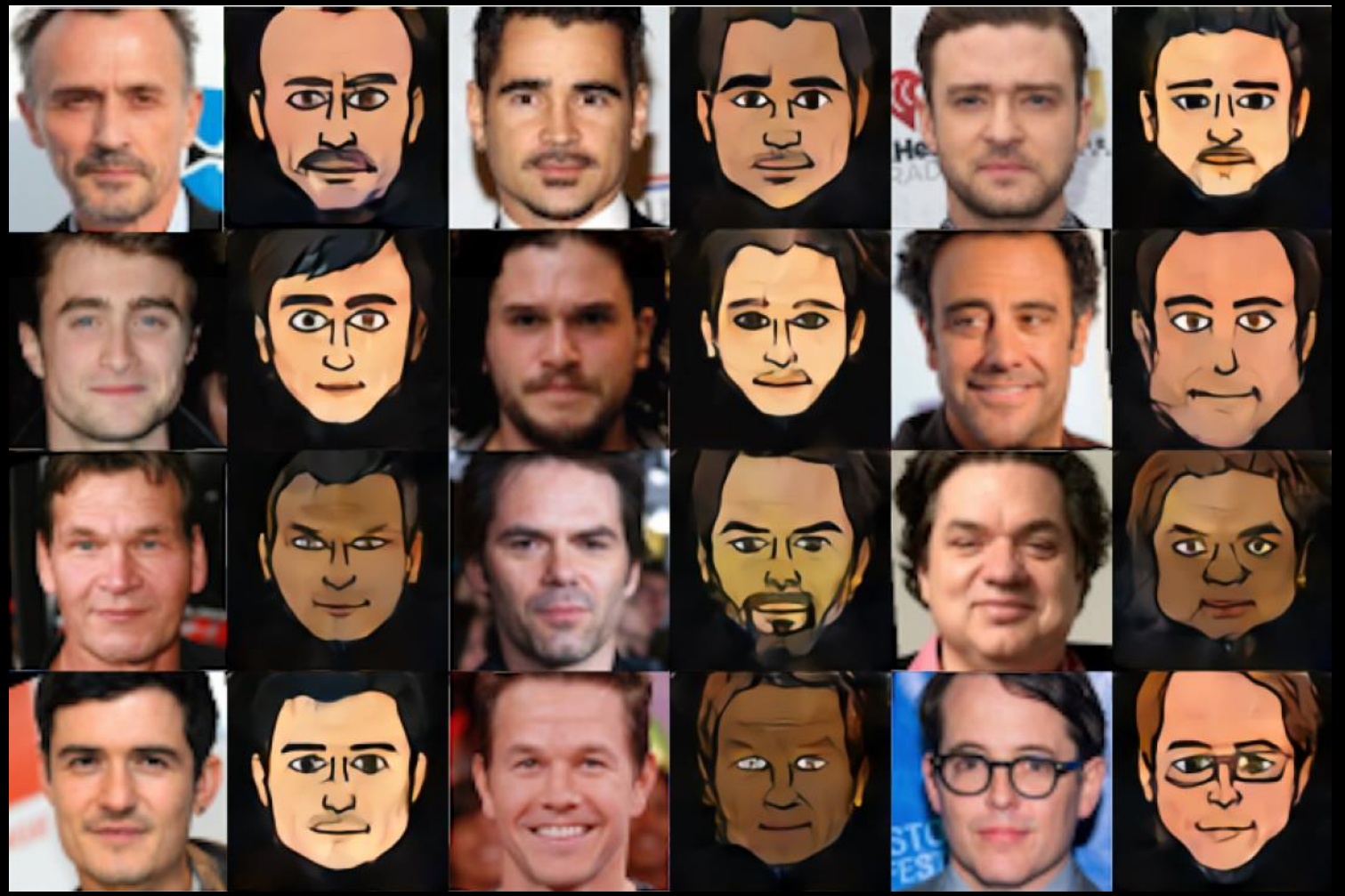
Transfer from SVHN to MNIST
(using Tensorboard)

Domain Transfer Network (DTN)



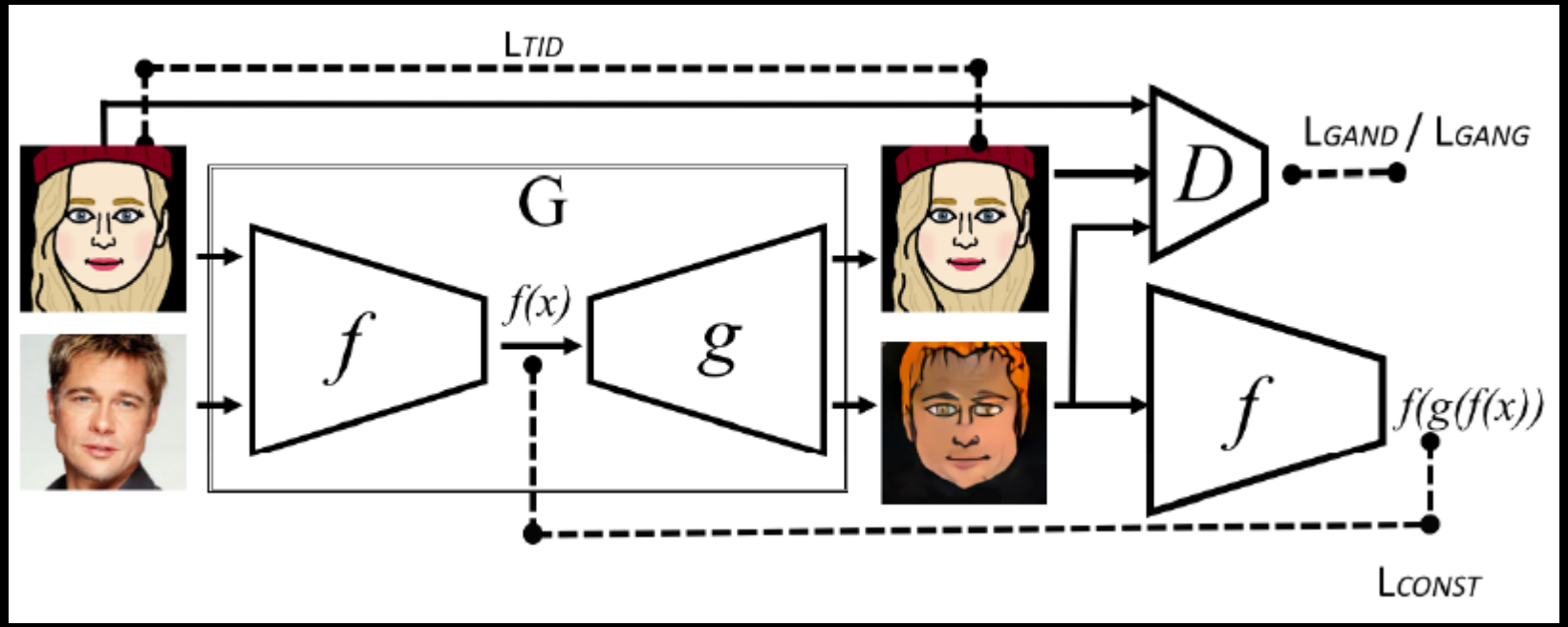
Transfer from SVHN to MNIST
(using Tensorboard)

Domain Transfer Network (DTN)



Transfer from face photos to emoji

Domain Transfer Network (DTN)



Domain Transfer Network (DTN)

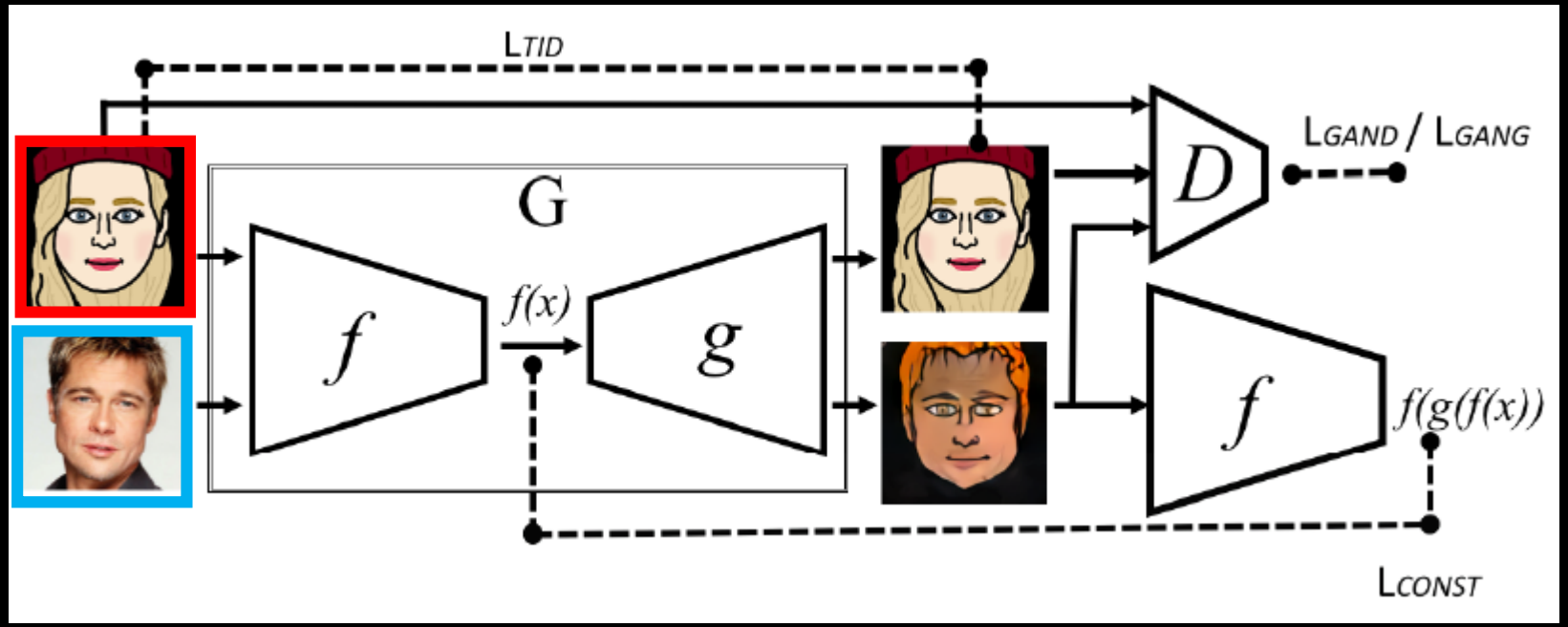
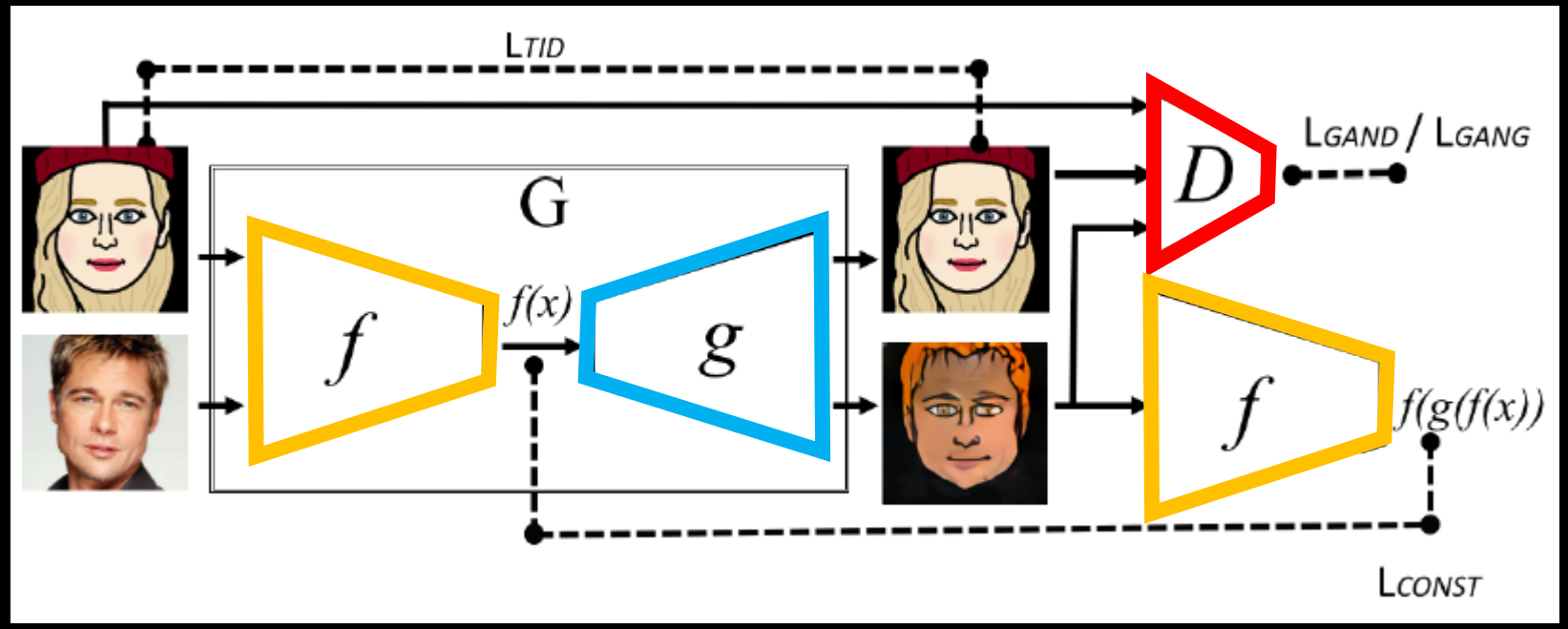


Image in target domain T

Image in source domain S

Domain Transfer Network (DTN)

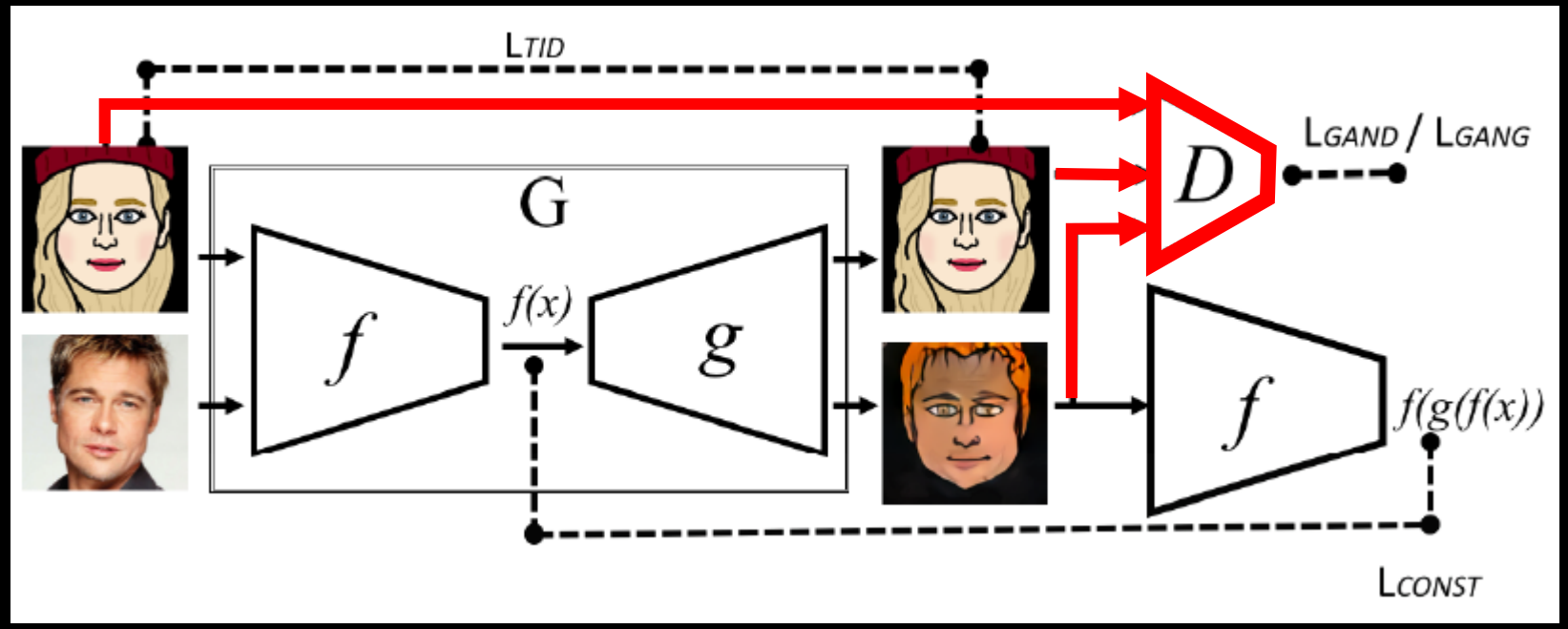


Discriminator

Generator

f-consistency term

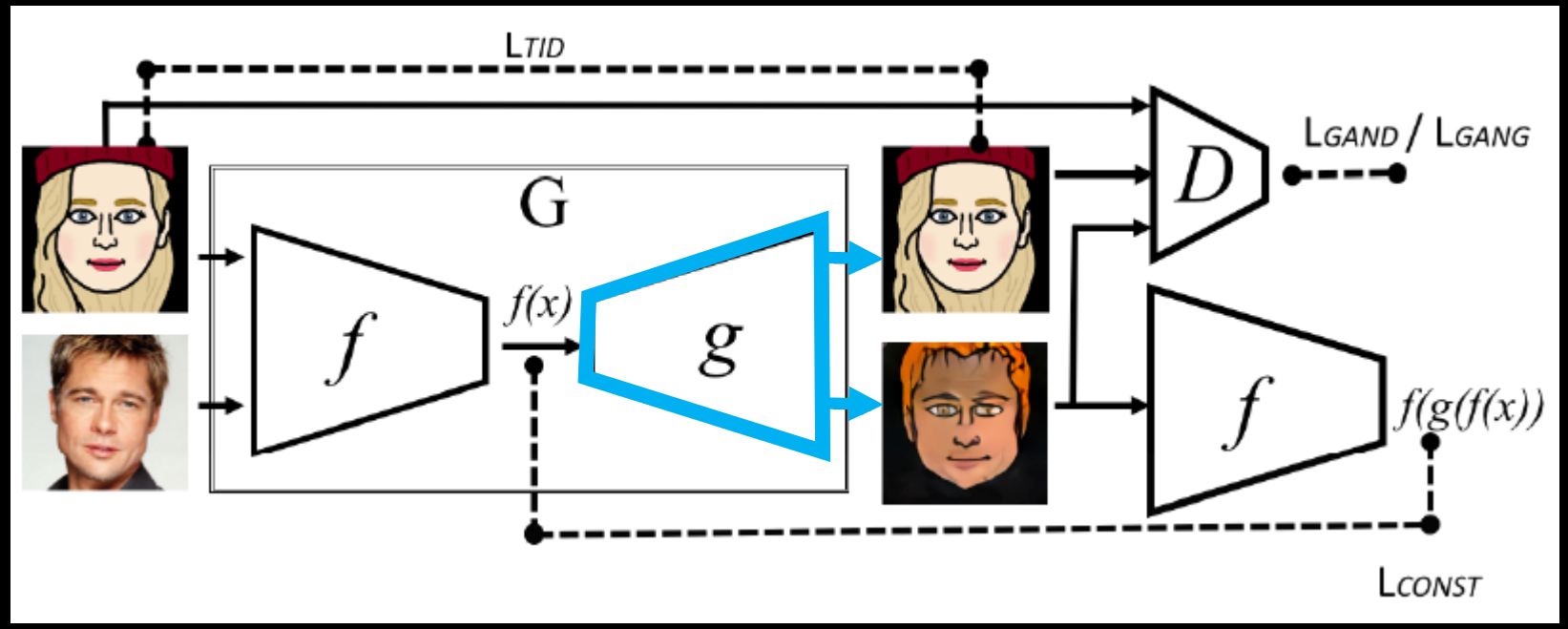
Domain Transfer Network (DTN)



Discriminator(식별자)

target image에 대해 진짜/가짜를 구별

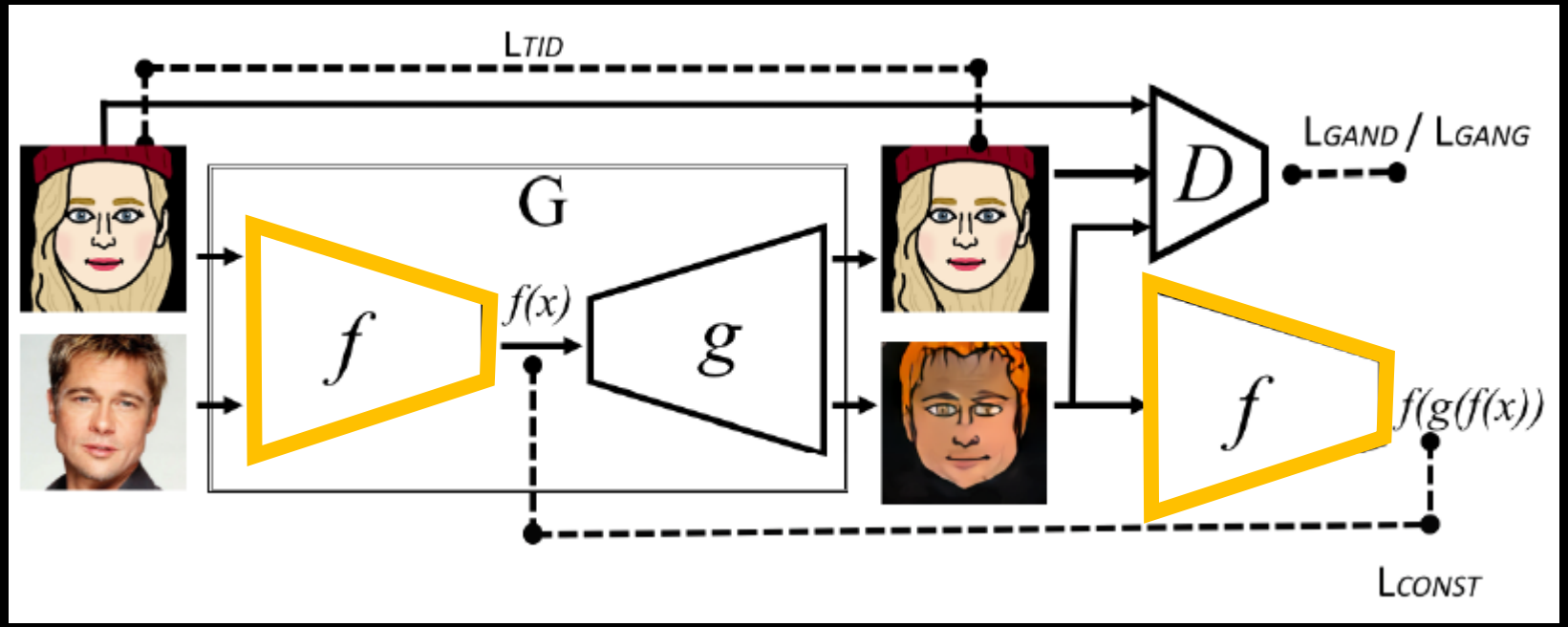
Domain Transfer Network (DTN)



Generator(생성자)

식별자를 속이기 위해 진짜 같은 가짜 이미지를 생성.
S -> T로 domain transfer

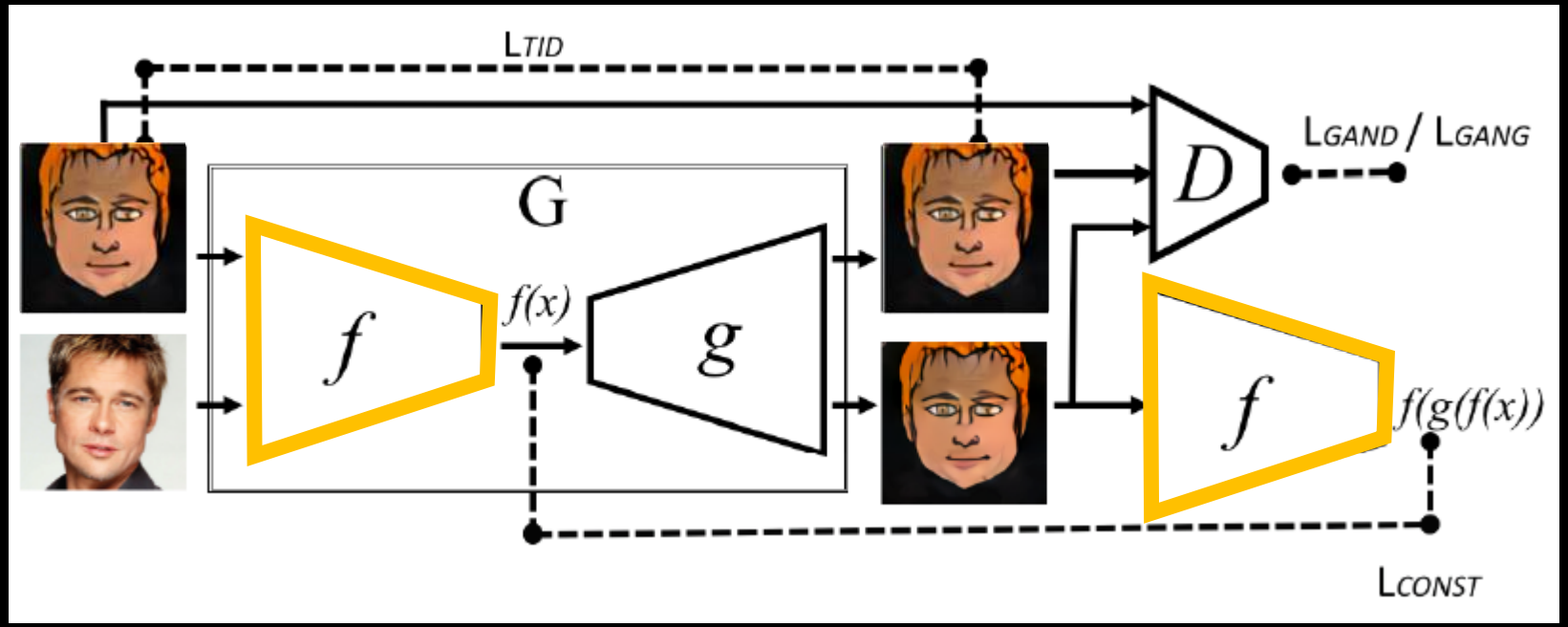
Domain Transfer Network (DTN)



f-consistency term

S->T로 비슷하게 domain transfer가 되게 함.

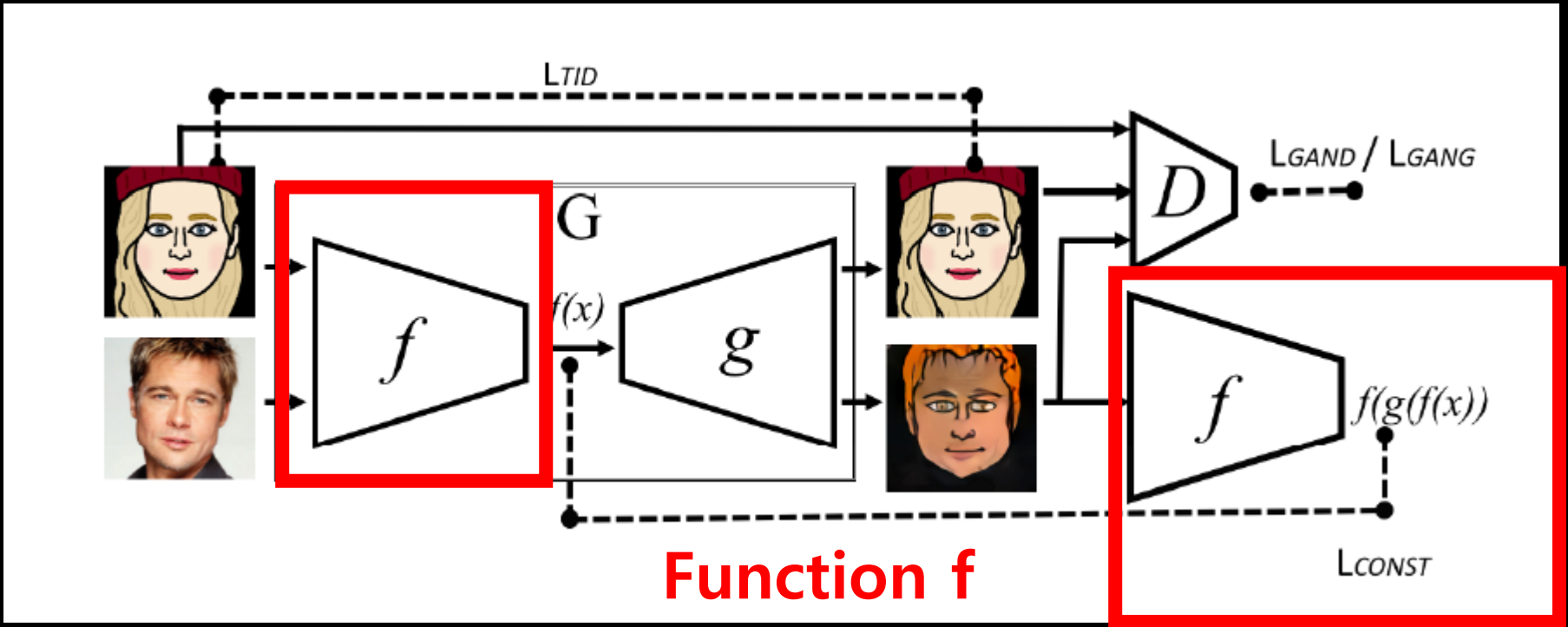
Domain Transfer Network (DTN)



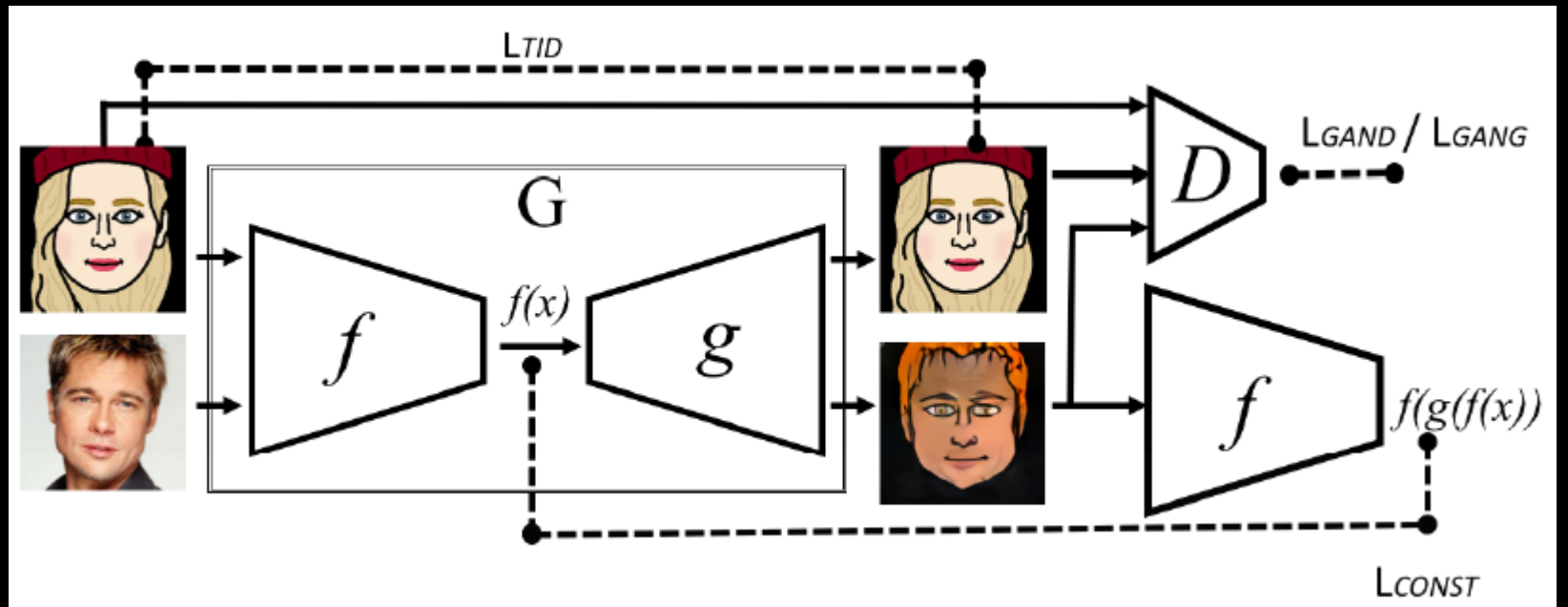
f-consistency term

S->T로 비슷하게 domain transfer가 되게 함.

Domain Transfer Network (DTN)



Domain Transfer Network (DTN)



$$L_{\mathrm{D}} = -\sum_{x \in \mathbf{s}} \log D_1(g(f(x))) - \sum_{x \in \mathbf{t}} \log D_2(g(f(x))) - \sum_{x \in \mathbf{t}} \log D_3(x) \quad (3)$$

$$L_{\mathrm{GANG}} = -\sum_{x \in \mathbf{s}} \log D_3(g(f(x))) - \sum_{x \in \mathbf{t}} \log D_3(g(f(x))) \quad (4)$$

$$L_{\mathrm{CONST}} = \sum_{x \in \mathbf{s}} d(f(x), f(g(f(x)))) \quad (5)$$

$$L_{\mathrm{TID}} = \sum_{x \in \mathbf{t}} d_2(x, G(x)) \quad (6)$$

Domain Transfer Network (DTN)

Table 1: Accuracy of the MNIST classifier on the sampled transferred by our DTN method from SHVN to MNIST.

Method	Accuracy
Baseline method (Sec. 3)	13.71%
DTN	90.66%
DTN w/0 L_{TID}	88.40%
DTN w/0 L_{CONST}	74.55%
DTN G does not contain f	36.90%
DTN w/0 L_{D} and L_{GANG}	34.70%
DTN w/0 L_{CONST} & L_{TID}	5.28%
Original SHVN image	40.06%

Domain Transfer Network (DTN)

Table 3: Comparison of recognition accuracy of the digit 3 as generated in MNIST

Method	Accuracy of '3'
DTN	94.67%
'3' was not shown in **s**	93.33%
'3' was not shown in **t**	40.13%
'3' was not shown in both **s** or **t**	60.02%
'3' was not shown in **s**, **t**, and during the training of f	4.52 %

Domain Transfer Network (DTN)

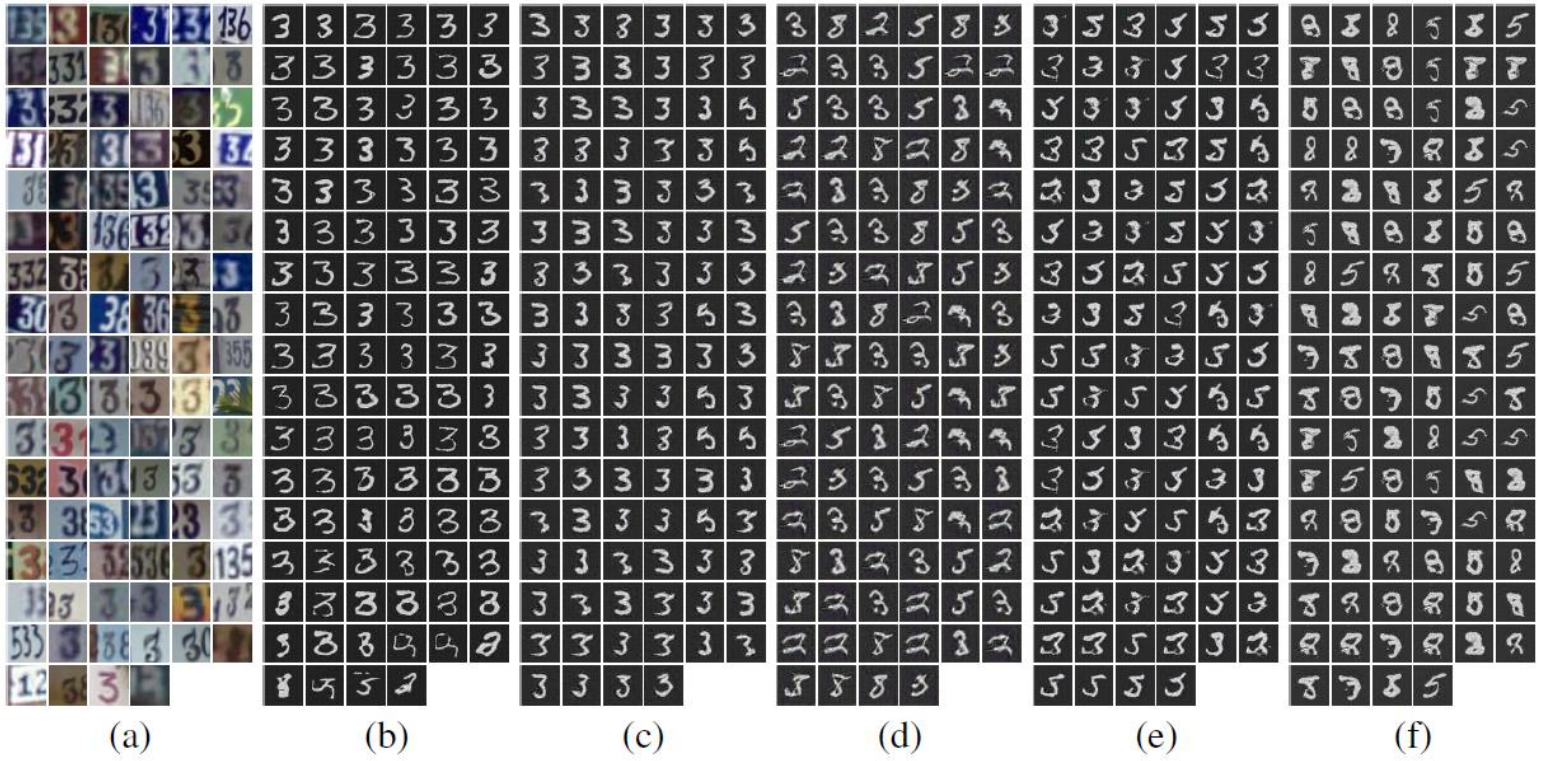


Figure 3: A random subset of the digit '3' from SVHN, transferred to MNIST. (a) The input images. (b) Results of our DTN. In all plots, the cases keep their respective locations, and are sorted by the probability of '3' as inferred by the MNIST classifier on the results of our DTN. (c) The obtained results, in which the digit 3 was not shown as part of the set $\mathbf{s}$ unlabeled samples from SVNH. (d) The obtained results, in which the digit 3 was not shown as part of the set $\mathbf{t}$ of unlabeled samples in MNIST. (e) The digit 3 was not shown in both $\mathbf{s}$ and $\mathbf{t}$. (f) The digit 3 was not shown in $\mathbf{s}$, $\mathbf{t}$, and during the training of f.

Domain Transfer Network (DTN)

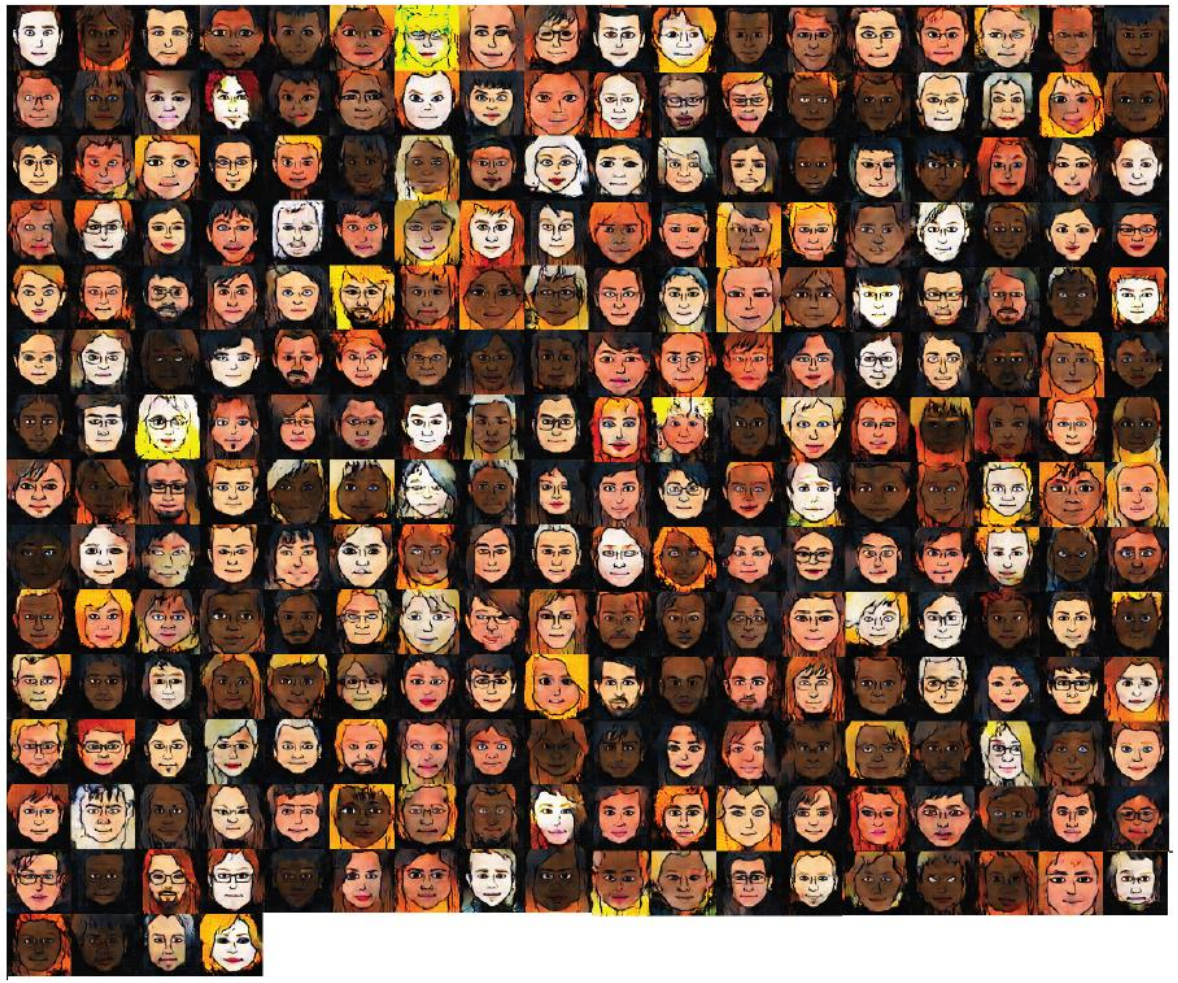


Figure 8: The emoji visualization of the standard basis vectors in the space of the face representation, i.e., $g(e_1)$,...,$g(e_{256})$, where e_i is the i standard basis vector in $\mathbb{R}^{256}$.

Domain Transfer Network (DTN)

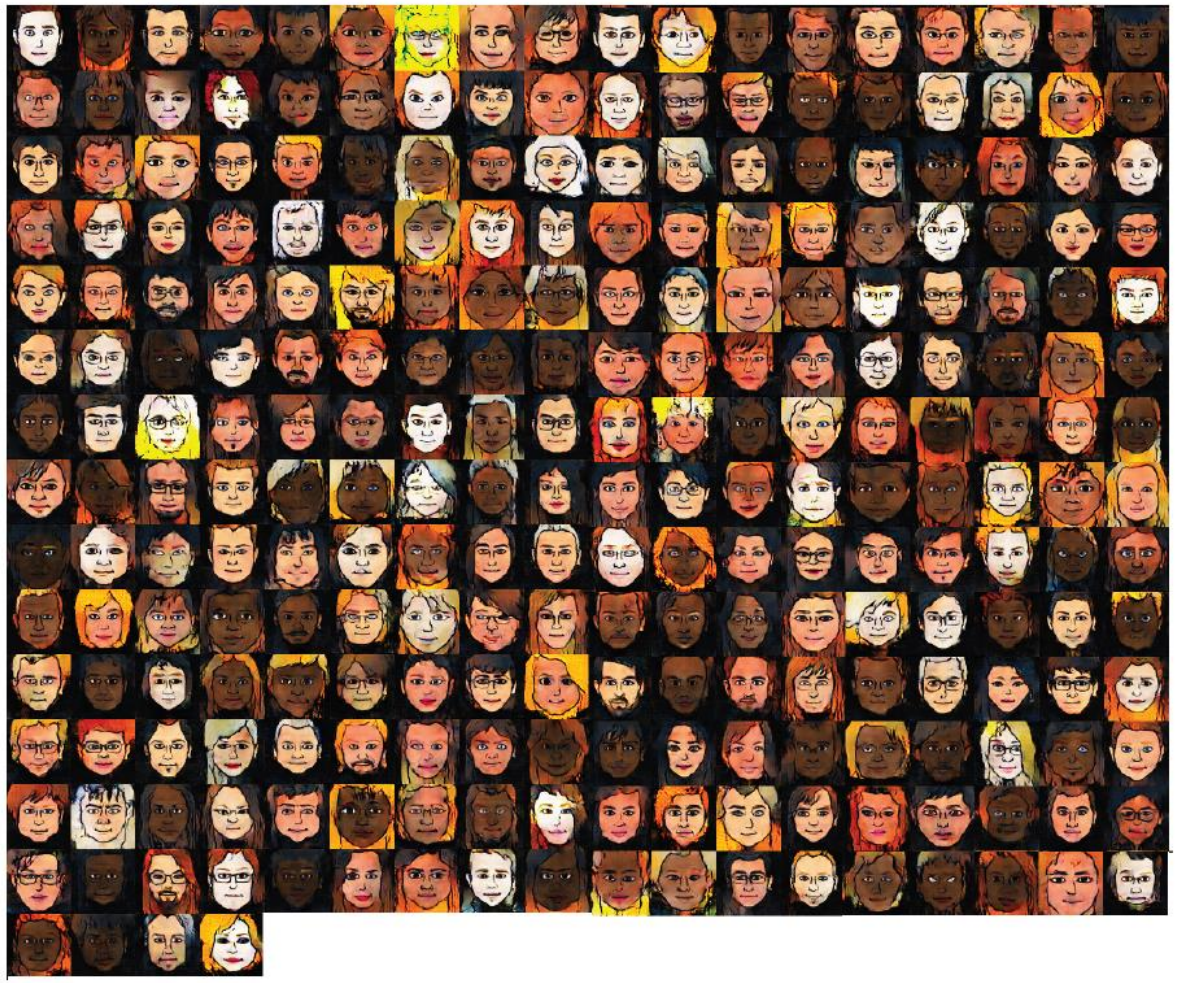
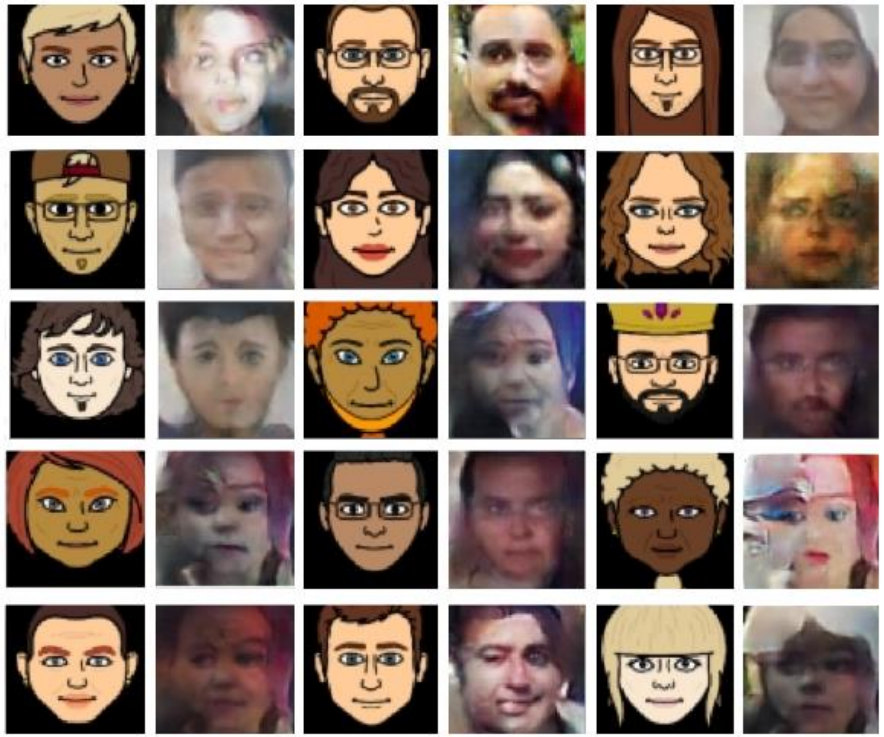


Figure 8: The emoji visualization of the standard basis vectors in the space of the face representation, i.e., $g(e_1)$,...,$g(e_{256})$, where e_i is the i standard basis vector in $\mathbb{R}^{256}$.

Domain Transfer Network (DTN)

(a)



(b)

Figure 9: Domain transfer in the other direction (see limitations in Sec. 6). Input (output) in odd (even) columns. (a) Transfer from MNIST to SVHN. (b) Transfer from emoji to face photos.