Introduction of Generative Adversarial Network (GAN)

Slide from:
H. Y. Lee http://speech.ee.ntu.edu.tw/~tlkagk/courses.html
https://www.youtube.com/watch?v=DQNNMiAP5lw

Generation

We will control what to generate latter. → Conditional Generation

Image Generation

$$\begin{bmatrix} 0.3 \\ -0.1 \\ \vdots \\ -0.7 \end{bmatrix} \begin{bmatrix} 0.1 \\ -0.1 \\ \vdots \\ 0.7 \end{bmatrix} \begin{bmatrix} -0.3 \\ 0.1 \\ \vdots \\ 0.9 \end{bmatrix} \rightarrow \boxed{\text{NN Generator}} \rightarrow$$

In a specific range

Sentence Generation

$$\begin{bmatrix} 0.3 \\ -0.1 \\ \vdots \\ -0.7 \end{bmatrix} \begin{bmatrix} 0.1 \\ -0.1 \\ \vdots \\ 0.2 \end{bmatrix} \begin{bmatrix} -0.3 \\ 0.1 \\ \vdots \\ 0.5 \end{bmatrix} \rightarrow \boxed{\text{NN Generator}} \rightarrow$$

How are you?

Good morning.

Good afternoon.

Powered by: http://mattya.github.io/chainer-DCGAN/

Basic Idea of GAN

It is a neural network (NN), or a function.

vector → Generator → image

high dimensional vector

$$\begin{bmatrix} 0.1 \\ -3 \\ \vdots \\ 2.4 \\ 0.9 \end{bmatrix}$$

→ Generator →

Each dimension of input vector represents some characteristics.

$$\begin{bmatrix} \boxed{3} \\ -3 \\ \vdots \\ 2.4 \\ 0.9 \end{bmatrix}$$

→ Generator →

Longer hair

$$\begin{bmatrix} 0.1 \\ 2.1 \\ \vdots \\ \boxed{5.4} \\ 0.9 \end{bmatrix}$$

→ Generator →

blue hair

$$\begin{bmatrix} 0.1 \\ -3 \\ \vdots \\ 2.4 \\ \boxed{3.5} \end{bmatrix}$$

→ Generator →

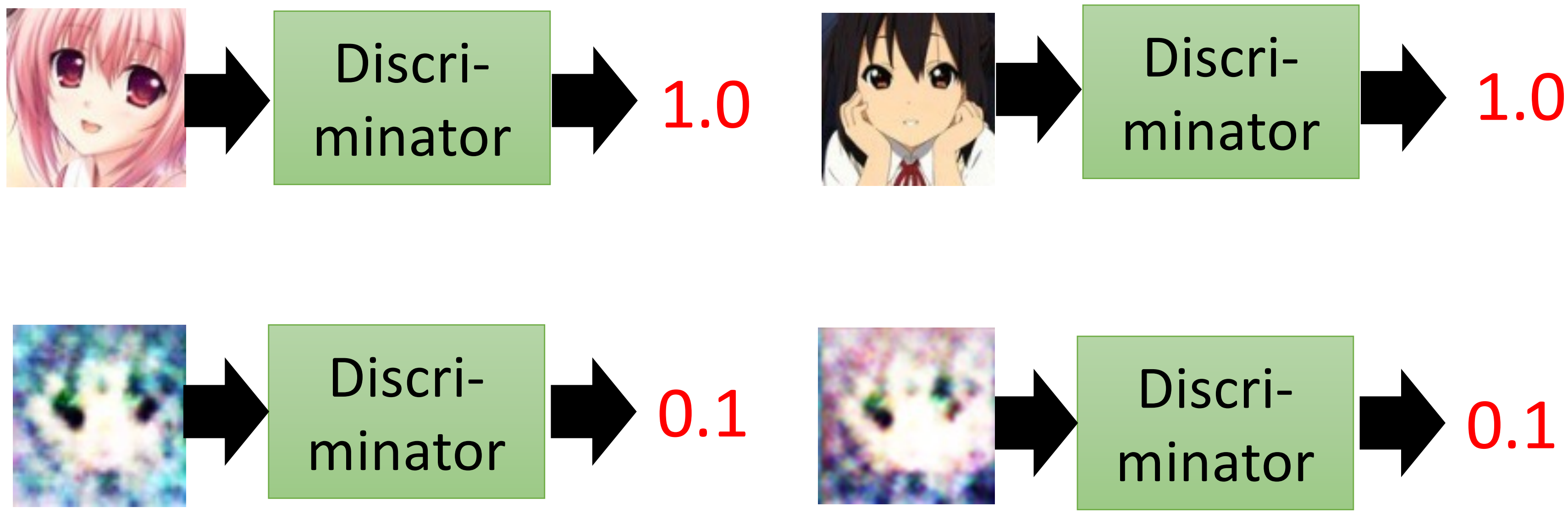
Open mouth

Basic Idea of GAN

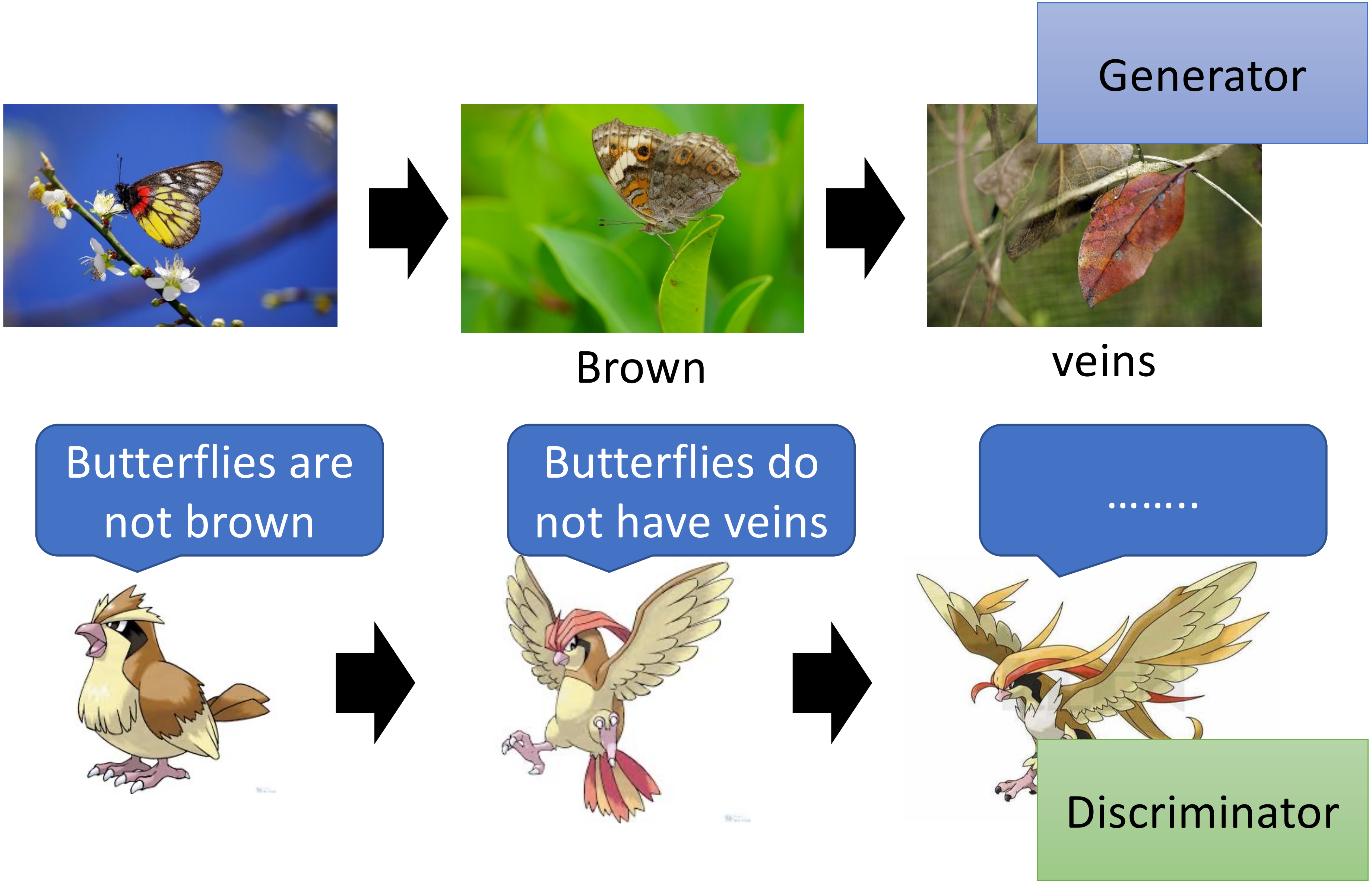
It is a neural network (NN), or a function.

image → Discri-minator → scalar

Larger value means real, smaller value means fake.



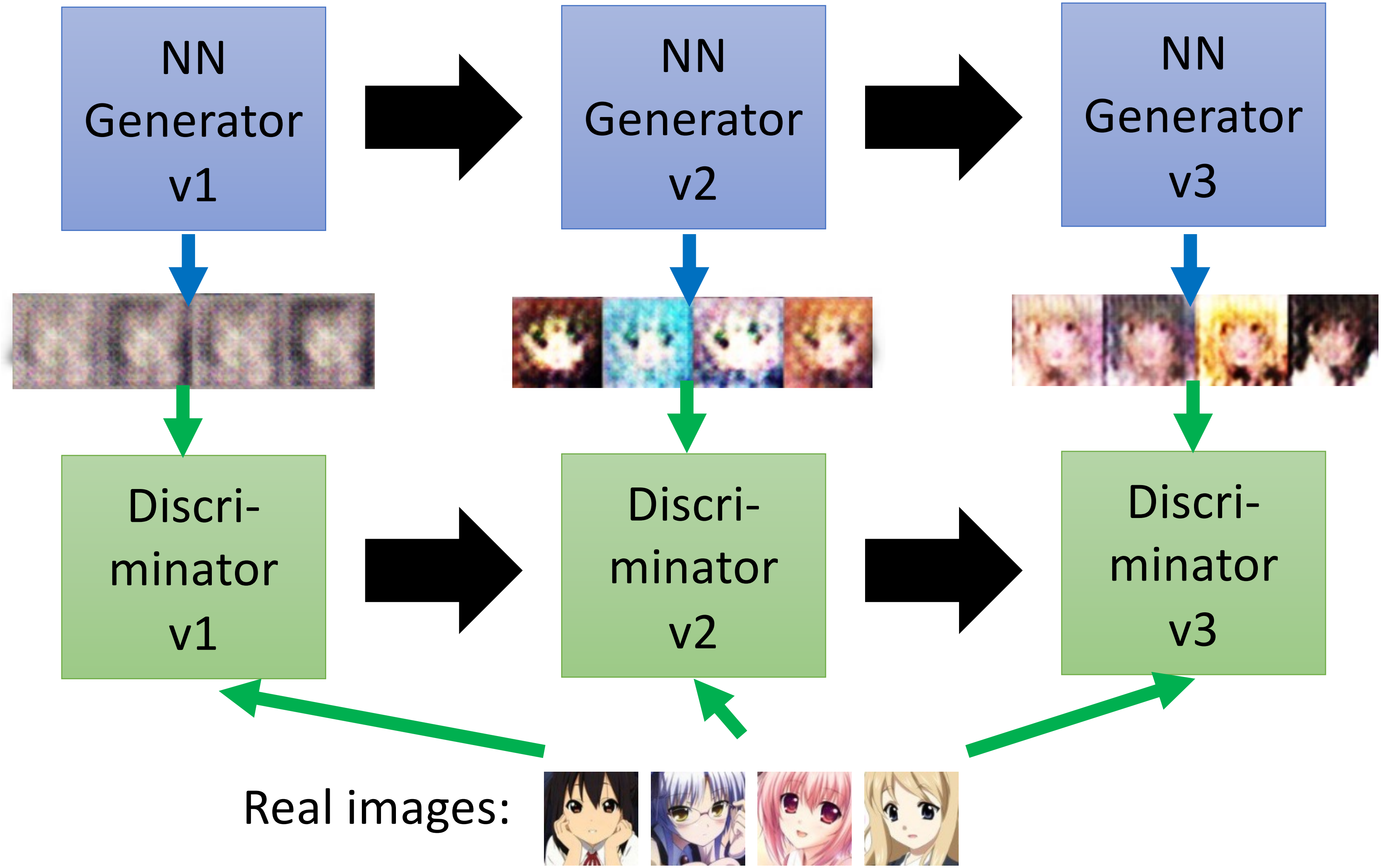
Basic Idea of GAN



Basic Idea of GAN

This is where the term "***adversarial***" comes from.

You can explain the process in different ways…….



Basic Idea of GAN (和平的比喻)

Generator (student)

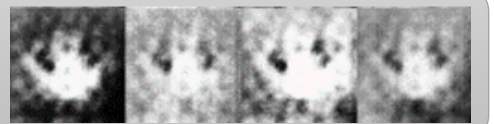
Discriminator (teacher)

Generator v1

Discriminator v1

沒有兩個圈

Generator v2



Discriminator v2

沒有彩色

Generator v3

為什麼不自己學？

為什麼不自己做？

Generator v.s. Discriminator

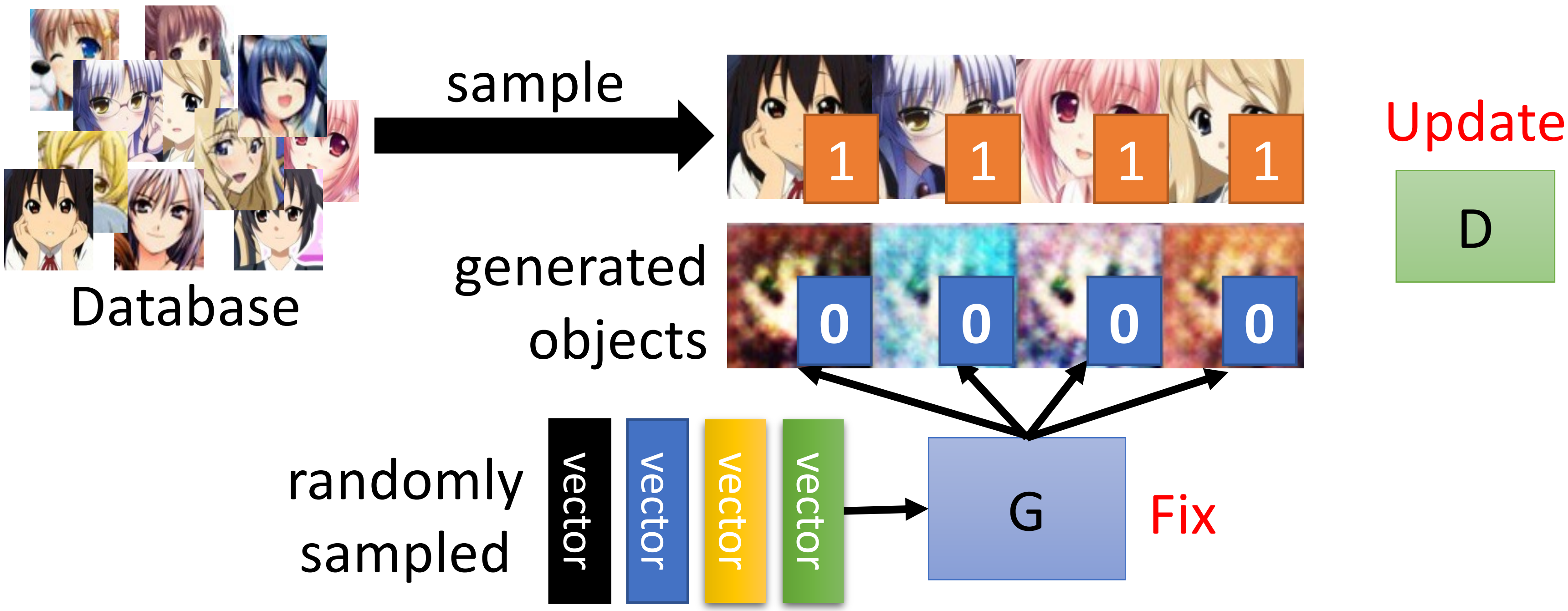
- 寫作敵人，唸做朋友

Algorithm

- Initialize generator and discriminator

- In each training iteration:

Step 1: Fix generator G, and update discriminator D



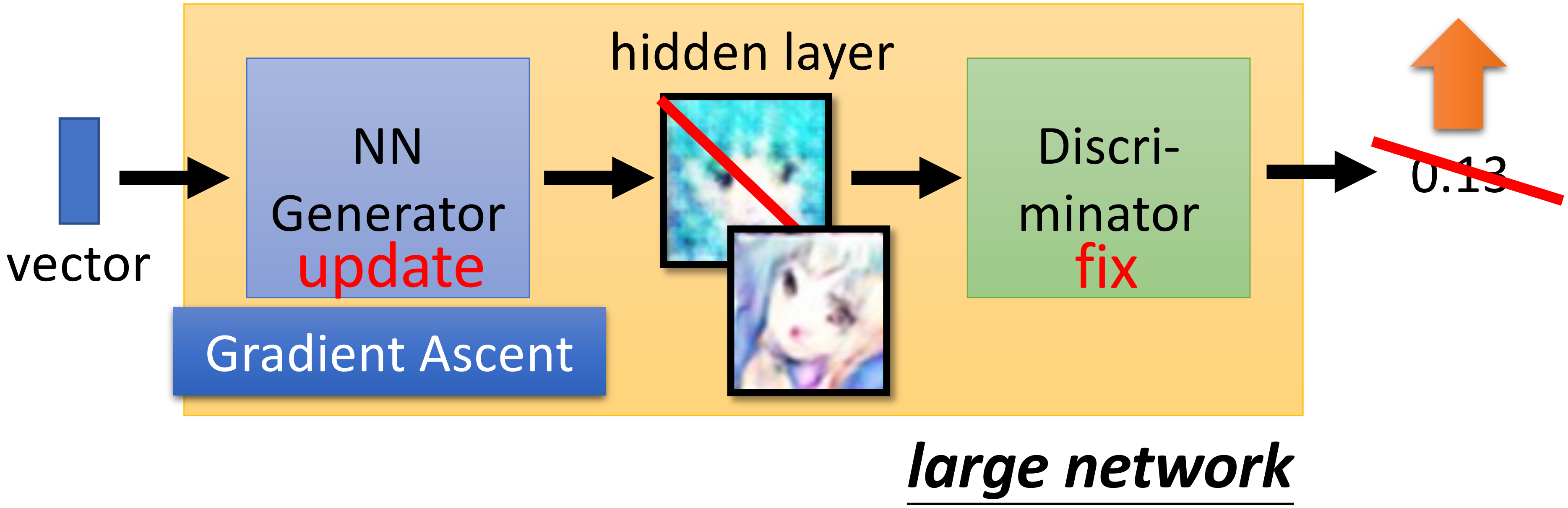
Discriminator learns to assign high scores to real objects and low scores to generated objects.

Algorithm

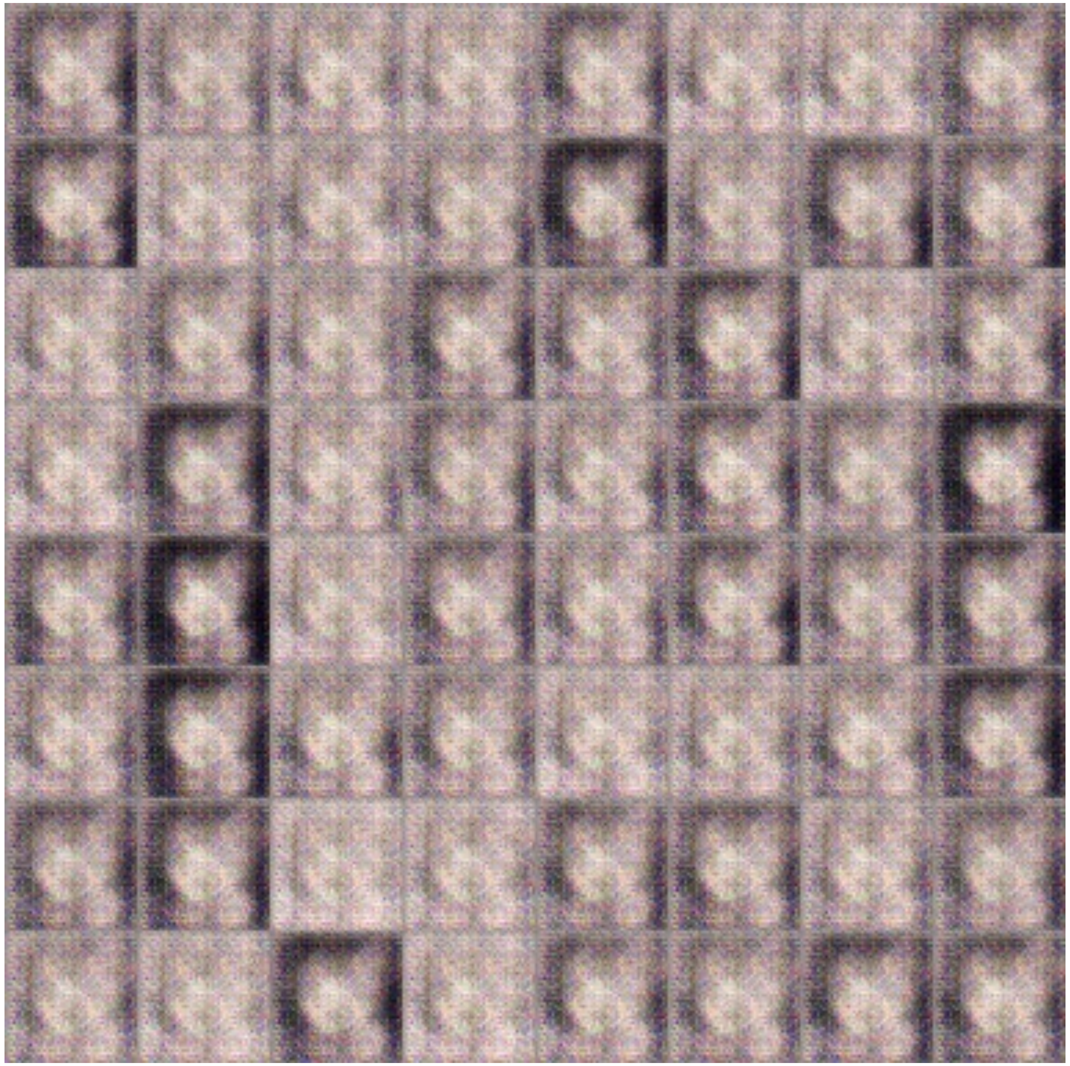
- Initialize generator and discriminator G D
- In each training iteration:

Step 2: Fix discriminator D, and update generator G

Generator learns to "fool" the discriminator



Anime Face Generation



100 updates

Source of training data: https://zhuanlan.zhihu.com/p/24767059

Anime Face Generation

1000 updates

Anime Face Generation

2000 updates

Anime Face Generation

5000 updates

Anime Face Generation

10,000 updates

Anime Face Generation

20,000 updates

Anime Face Generation

50,000 updates

Basic Idea of GAN
(和平的比喻)

Generator
(student)

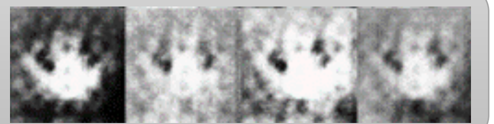
Discriminator
(teacher)

Generator
v1

Discriminator
v1

沒有兩個圈

Generator
v2



Discriminator
v2

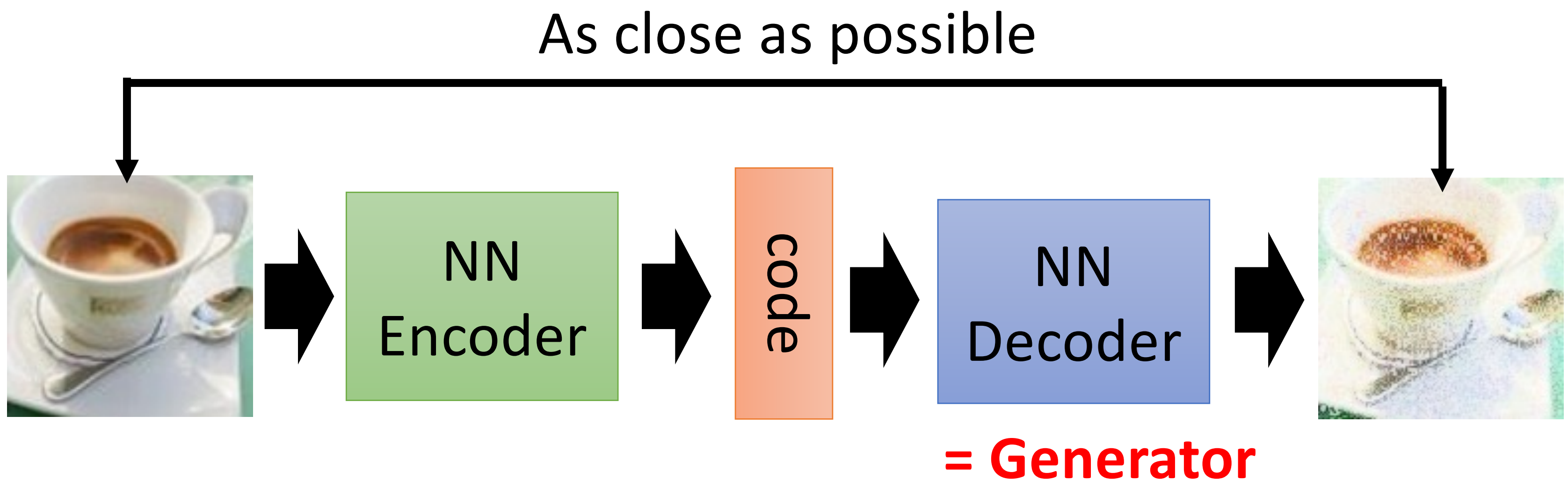
沒有彩色

Generator
v3

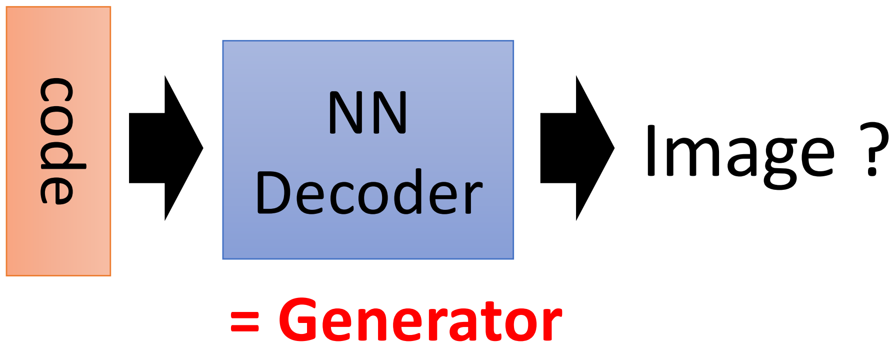
為什麼不自己學？

為什麼不自己做？

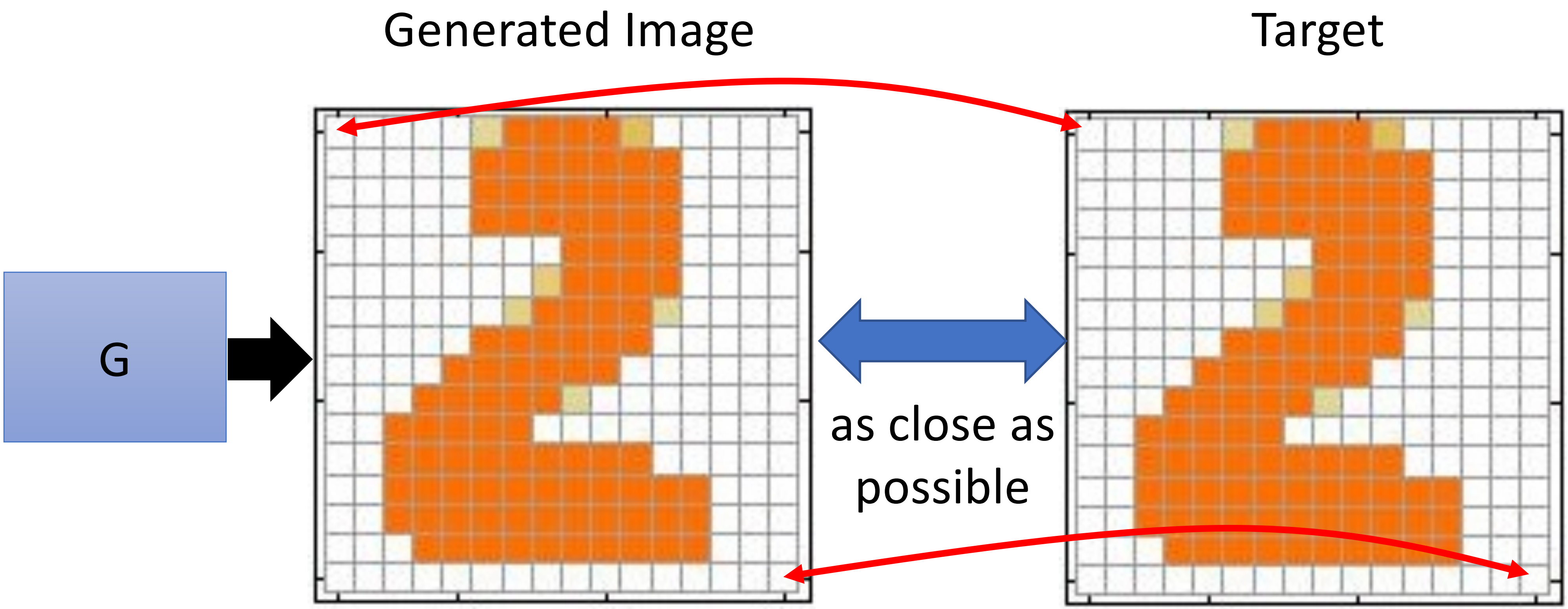
Auto-encoder



Randomly generate
a vector as code



What do we miss?



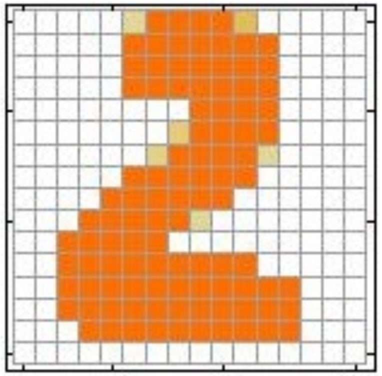
It will be fine if the generator can truly copy the target image.

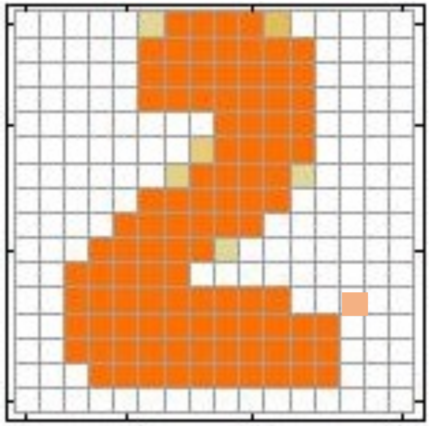
What if the generator makes some mistakes …….

Some mistakes are serious, while some are fine.

What do we miss?

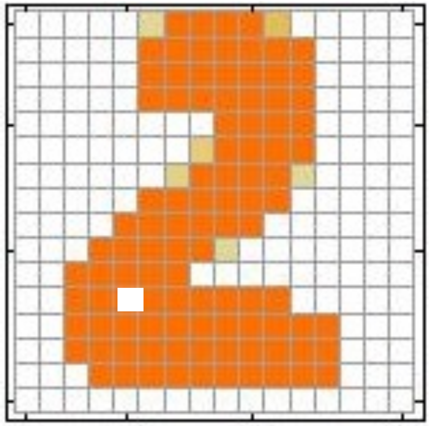
Target





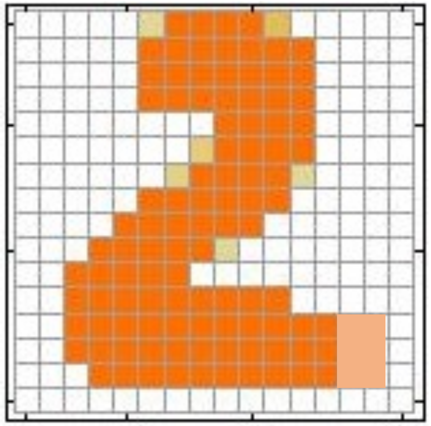
1 pixel error

我覺得不行



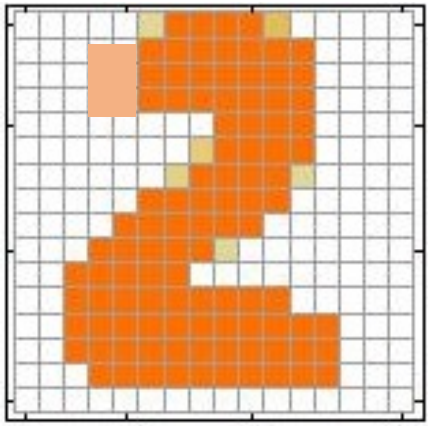
1 pixel error

我覺得不行



6 pixel errors

我覺得其實
可以



6 pixel errors

我覺得其實
可以

Discriminator

Evaluation function, Potential Function, Energy Function …

- Discriminator is a function D (network, can deep)

$$\mathrm{D}: X \rightarrow \mathrm{R}$$

  - Input x: an object x (e.g. an image)
  - Output D(x): scalar which represents how “good” an object x is

Can we use the discriminator to generate objects?

Yes.

Discriminator

- Suppose we already have a good discriminator D(x) ...

Inference

- Generate object $\tilde{x}$ that
$$\tilde{x} = \arg\max_{x \in X} D(x)$$

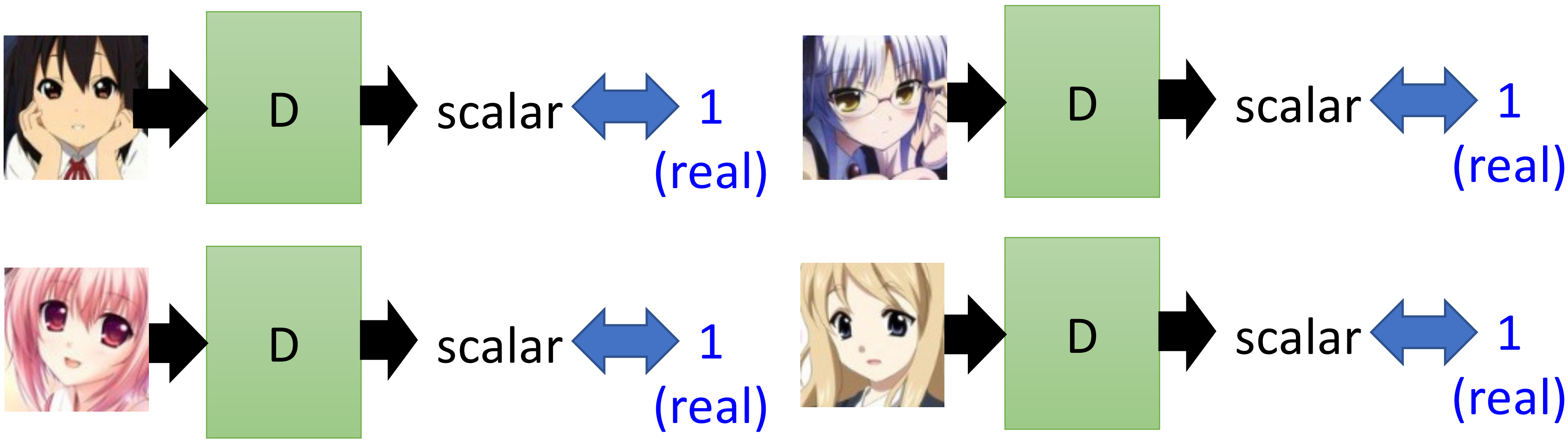
Enumerate all possible x !!!

It is feasible ???

How to learn the discriminator?

Discriminator - Training

- I have some real images

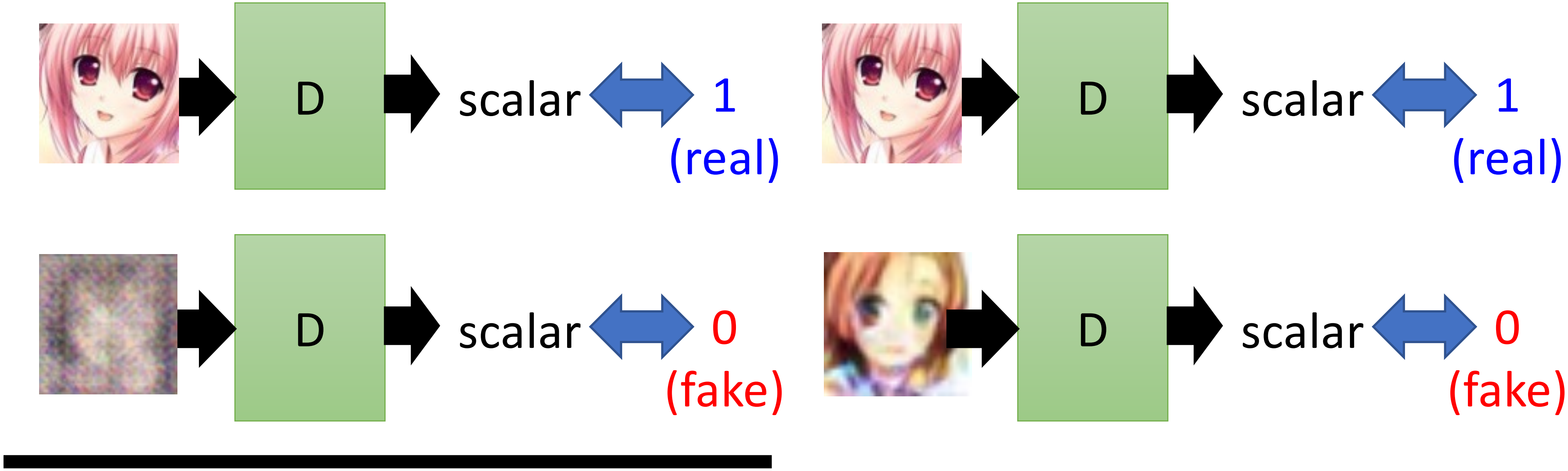


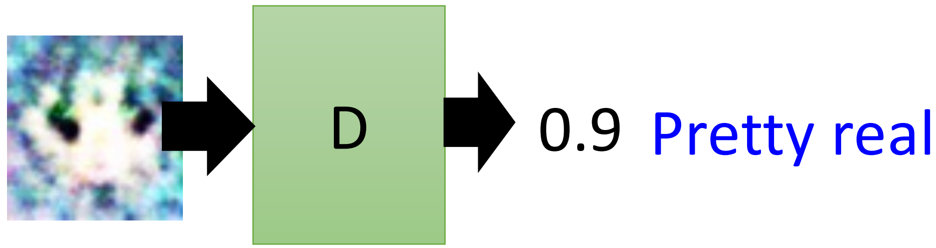
Discriminator only learns to output “1” (real).

Discriminator training needs some negative examples.

Discriminator - Training

- Negative examples are critical.



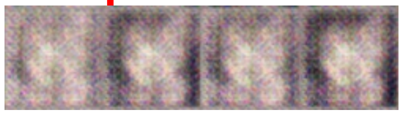


How to generate realistic negative examples?

Discriminator - Training

- **<u>General Algorithm</u>**

  - Given a set of positive examples, randomly generate a set of negative examples.



  - In each iteration
    - Learn a discriminator D that can discriminate positive and negative examples.

 v.s.  → D

    - Generate negative examples by discriminator D

$$\tilde{x} = \arg\max_{x \in X} D(x)$$

Generator v.s. Discriminator

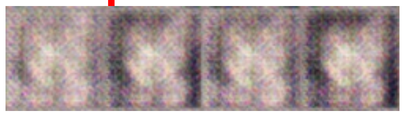
- ***Generator***
- Pros:
  - Easy to generate even with deep model
- Cons:
  - Imitate the appearance
  - Hard to learn the correlation between components

- ***Discriminator***
- Pros:
  - Considering the big picture
- Cons:
  - Generation is not always feasible
    - Especially when your model is deep
  - How to do negative sampling?

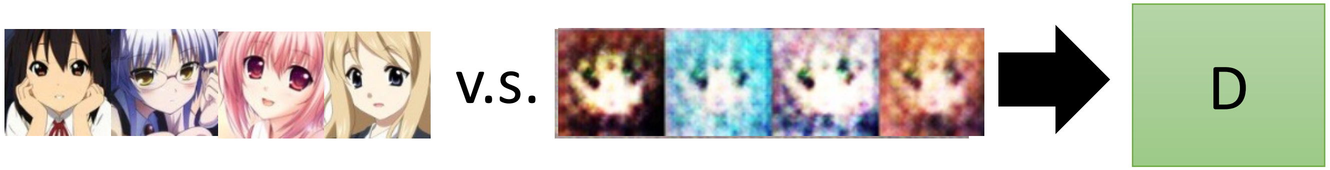
Generator + Discriminator

- **General Algorithm**

  - Given a set of positive examples, randomly generate a set of negative examples.



  - In each iteration
    - Learn a discriminator D that can discriminate positive and negative examples.

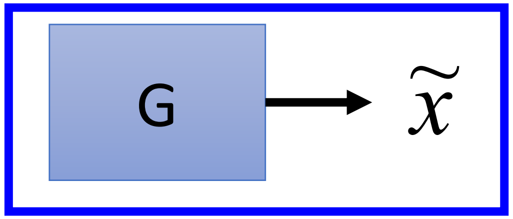


    - Generate negative examples by discriminator D

$$\boxed{G \rightarrow \tilde{x}} = \boxed{\tilde{x} = \arg\max_{x \in X} D(x)}$$

Benefit of GAN

- From Discriminator's point of view
  - Using generator to generate negative samples



=

$$\tilde{x} = \arg\max_{x \in X} D(x)$$

efficient

- From Generator's point of view
  - Still generate the object component-by-component
  - But it is learned from the discriminator with global view.

Introduction of Generative Adversarial Network (GAN)

Slide from:
H. Y. Lee http://speech.ee.ntu.edu.tw/~tlkagk/courses.html
https://www.youtube.com/watch?v=DQNNMiAP5lw