Introduction of Generative Adversarial Network (GAN)

李宏毅

Hung-yi Lee

Yann LeCun’s comment

What are some recent and potentially upcoming breakthroughs in unsupervised learning?

Yann LeCun, Director of AI Research at Facebook and Professor at NYU

Written Jul 29 · Upvoted by Joaquin Quiñonero Candela, Director Applied Machine Learning at Facebook and Huang Xiao

Adversarial training is the coolest thing since sliced bread.

I’ve listed a bunch of relevant papers in a previous answer.

Expect more impressive results with this technique in the coming years.

What’s missing at the moment is a good understanding of it so we can make it work reliably. It’s very finicky. Sort of like ConvNet were in the 1990s, when I had the reputation of being the only person who could make them work (which wasn’t true).

https://www.quora.com/What-are-some-recent-and-potentially-upcoming-breakthroughs-in-unsupervised-learning

Yann LeCun’s comment

What are some recent and potentially upcoming breakthroughs in deep learning?

Yann LeCun, Director of AI Research at Facebook and Professor at NYU

Written Jul 29 · Upvoted by Joaquin Quiñonero Candela, Director Applied Machine Learning at Facebook and Nikhil Garg, I lead a team of Quora engineers working on ML/NLP problems

……

The most important one, in my opinion, is adversarial training (also called GAN for Generative Adversarial Networks). This is an idea that was originally proposed by Ian Goodfellow when he was a student with Yoshua Bengio at the University of Montreal (he since moved to Google Brain and recently to OpenAI).

This, and the variations that are now being proposed is the most interesting idea in the last 10 years in ML, in my opinion.

https://www.quora.com/What-are-some-recent-and-potentially-upcoming-breakthroughs-in-deep-learning

Generative Adversarial Network (GAN)

- How to pronounce “GAN”?

Google 小姐

Outline

Basic Idea of GAN

When do we need GAN?

GAN as structured learning algorithm

Conditional Generation by GAN

- Modifying input code
- Paired data
- Unpaired data
- Application: Intelligent Photoshop

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

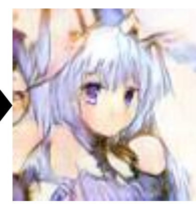
Basic Idea of GAN

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

vector → Generator → image

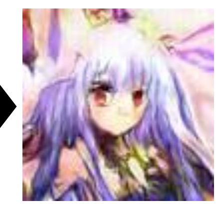
matrix

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



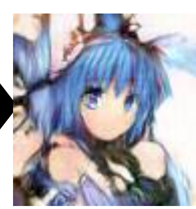
Each dimension of input vector represents some characteristics.

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



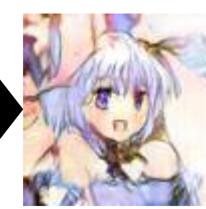
Longer hair

$$\begin{bmatrix} 0.1 \\ 2.1 \\ \vdots \\ \boxed{2.4} \\ 0.9 \end{bmatrix}$$ → Generator →



blue hair

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



Open mouth

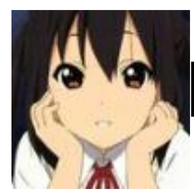
Basic Idea of GAN

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

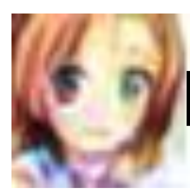
image → Discri-minator → scalar

Larger value means real, smaller value means fake.

 → Discri-minator → 1.0

 → Discri-minator → 1.0

 → Discri-minator → 0.1

 → Discri-minator → ?

Basic Idea of GAN

Generator

Brown

veins

Butterflies are not brown

Butterflies do not have veins

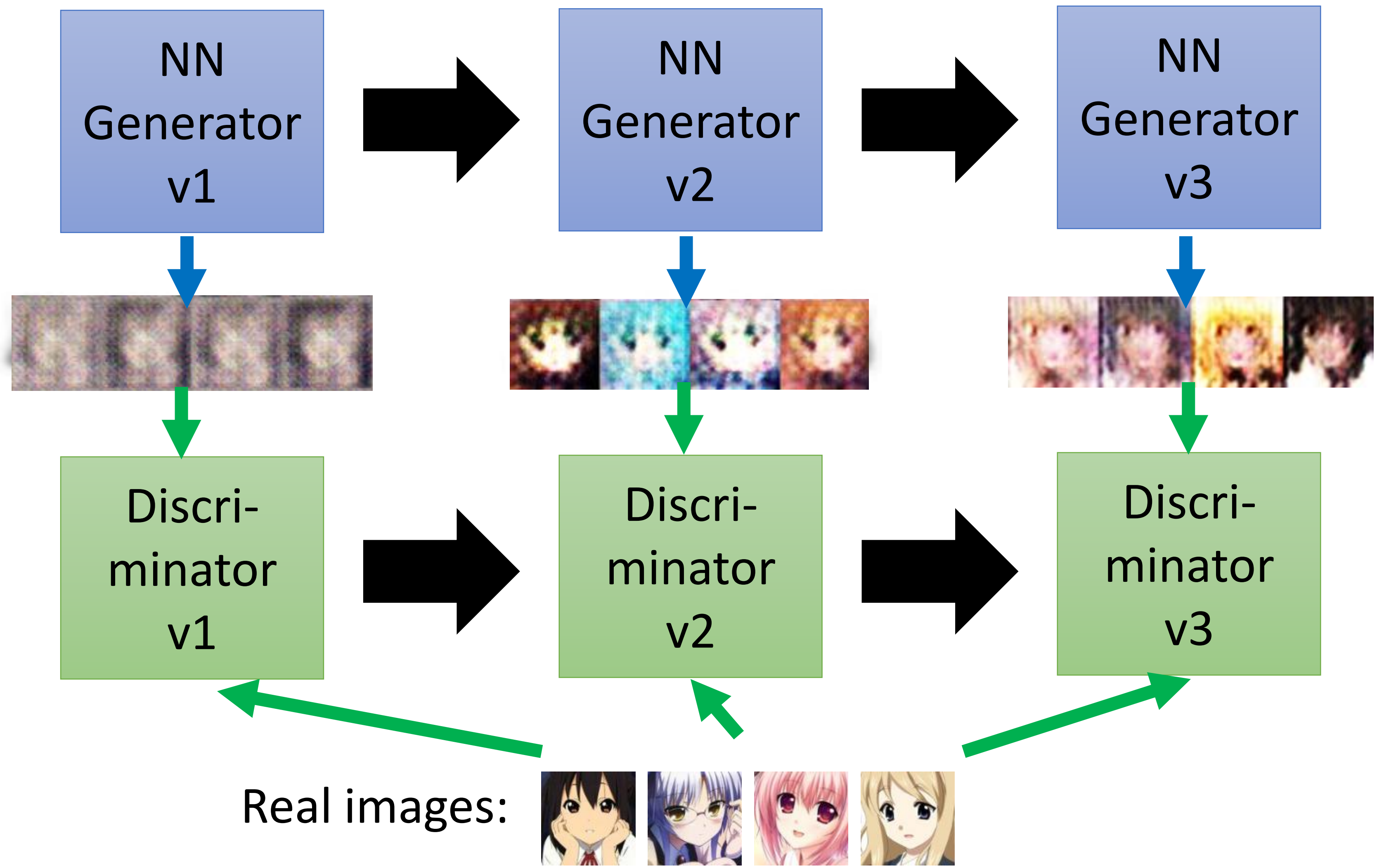
……..

Discriminator

Basic Idea of GAN

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

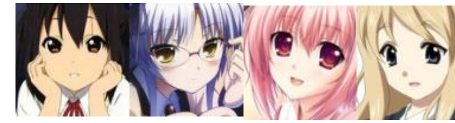
You can explain the process in different ways…….



Basic Idea of GAN (和平的比喻)

Generator (student)

Discriminator (teacher)



Generator v1

Discriminator v1

No eyes

Generator v2

Discriminator v2

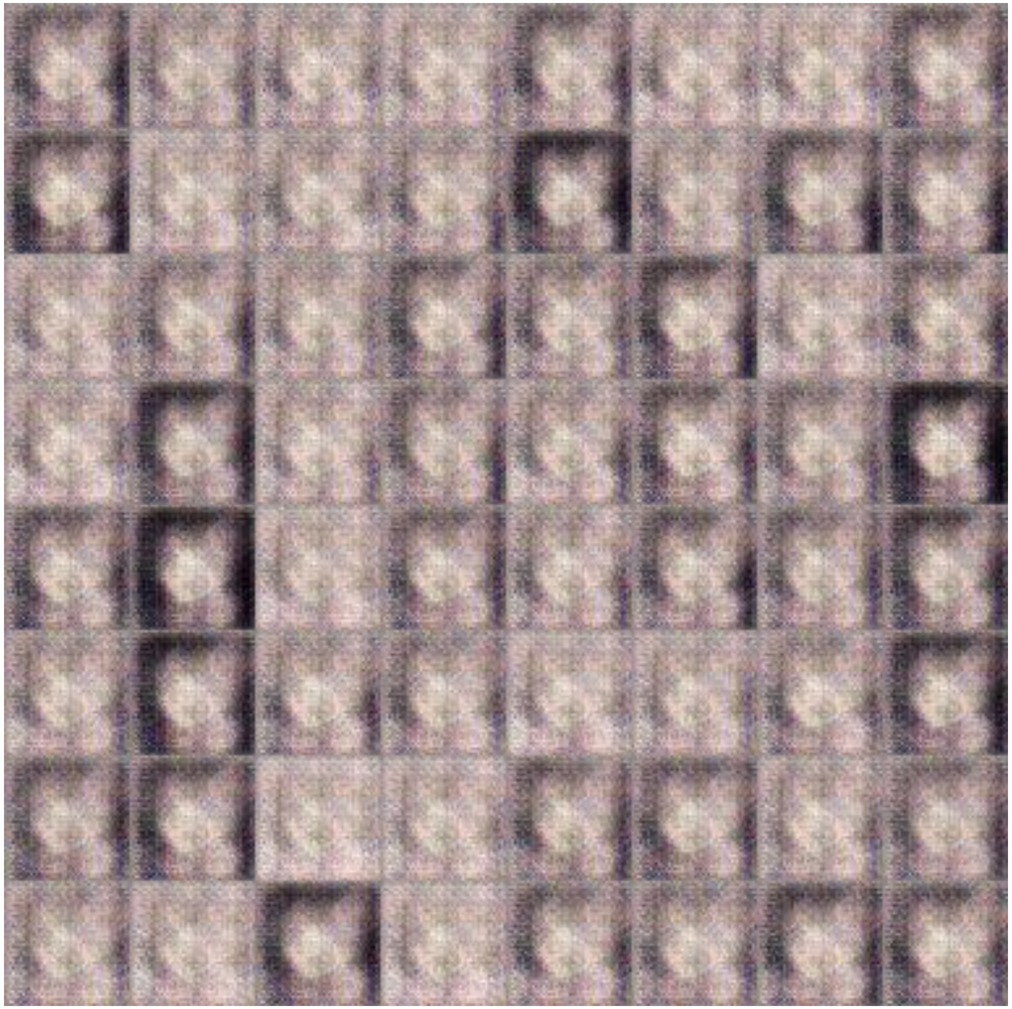
No mouth

Generator v3

為什麼不自己學？

為什麼不自己做？

Anime Face Generation



100 updates

Anime Face Generation

1000 updates

Anime Face Generation

2000 updates

Anime Face Generation

5000 updates

Anime Face Generation

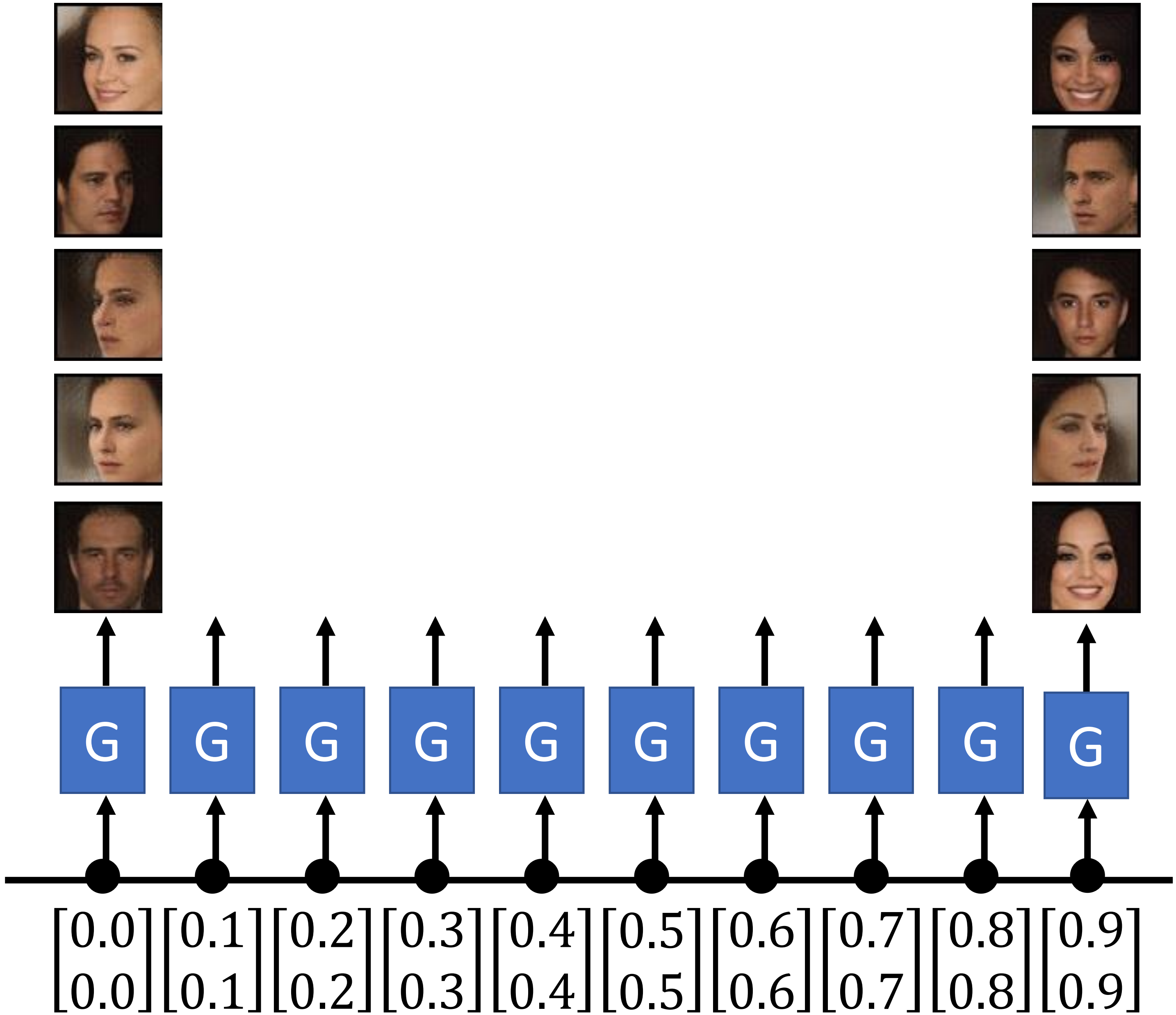
10,000 updates

Anime Face Generation

20,000 updates

Anime Face Generation

50,000 updates



感謝陳柏文同學提供實驗結果

Outline

Basic Idea of GAN

When do we need GAN?

GAN as structured learning algorithm

Conditional Generation by GAN

- Modifying input code
- Paired data
- Unpaired data
- Application: Intelligent Photoshop

Structured Learning

Machine learning is to find a function f

$$f : X \rightarrow Y$$

Regression: output a scalar

Classification: output a “class” (one-hot vector)

1	0	0

Class 1

0	1	0

Class 2

0	0	1

Class 3

Structured Learning/Prediction: output a sequence, a matrix, a graph, a tree ……

Output is composed of components with dependency

GAN

Regression,
Classification

Output Sequence $f: X \rightarrow Y$

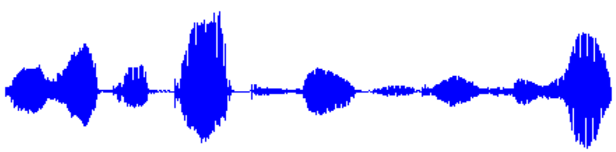
Machine Translation

X : “機器學習及其深層與結構化”
(sentence of language 1)

Y : “Machine learning and having it deep and structured”
(sentence of language 2)

Speech Recognition

X :



(speech)

Y : 感謝大家來上課”
(transcription)

Chat-bot

X : “How are you?”
(what a user says)

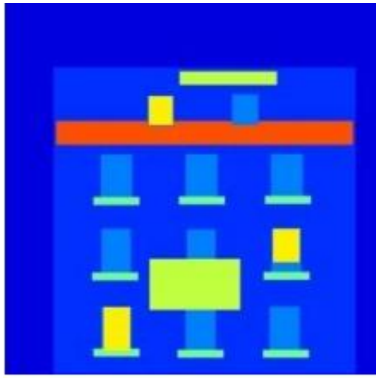
Y : “I’m fine.”
(response of machine)

Output Matrix

$$f : X \rightarrow Y$$

Image to Image

X :



Y :

Colorization:

Ref: https://arxiv.org/pdf/1611.07004v1.pdf

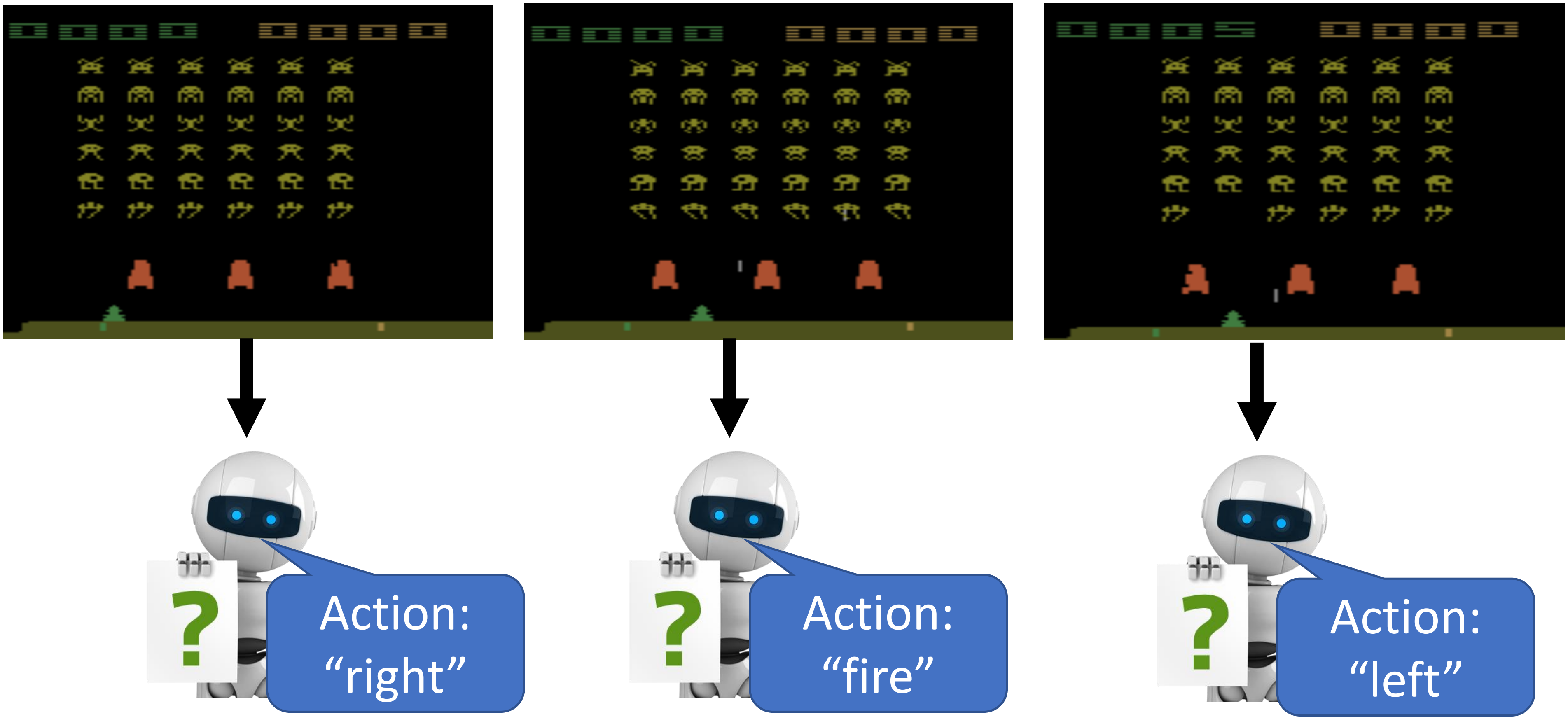
Text to Image

X : "this white and yellow flower have thin white petals and a round yellow stamen"

Y :

ref: https://arxiv.org/pdf/1605.05396.pdf

Decision Making and Control



A sequence of decisions

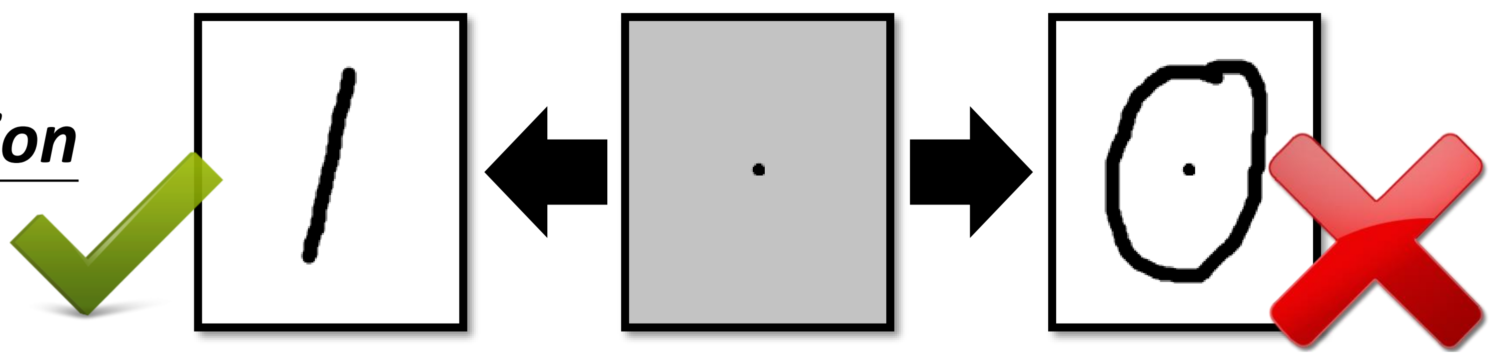
Why Structured Learning Interesting?

- **One-shot/Zero-shot Learning**:
  - In classification, each class has some examples.
  - In structured learning,
    - If you consider each possible output as a “class” ……
    - Since the output space is huge, most “classes” do not have any training data.
    - Machine has to create new stuff during testing.
    - Need more intelligence

Why Structured Learning Interesting?

- Machine has to learn to ***planning***
  - Machine can generate objects component-by-component, but it should have a big picture in its mind.
  - Because the output components have dependency, they should be considered globally.

Image Generation



Sentence Generation

這個婆娘不是人

九天玄女下凡塵

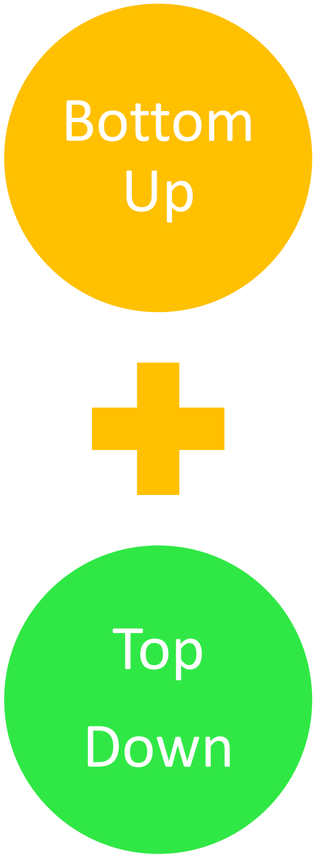
Structured Learning Approach

Generator

Learn to generate the object at the component level

Discriminator

Evaluating the whole object, and find the best one



Outline

Basic Idea of GAN

When do we need GAN?

GAN as structured learning algorithm

Conditional Generation by GAN

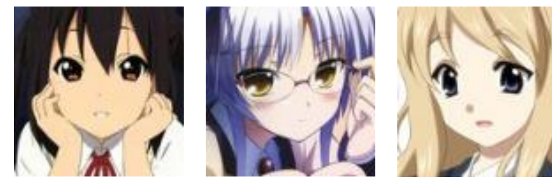
- Modifying input code
- Paired data
- Unpaired data
- Application: Intelligent Photoshop

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?

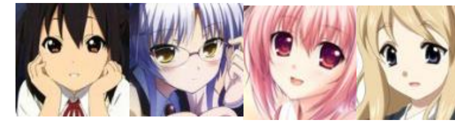
Good morning.

Good afternoon.

Basic Idea of GAN (和平的比喻)

Generator (student)

Discriminator (teacher)



Generator v1

Discriminator v1

No eyes

Generator v2

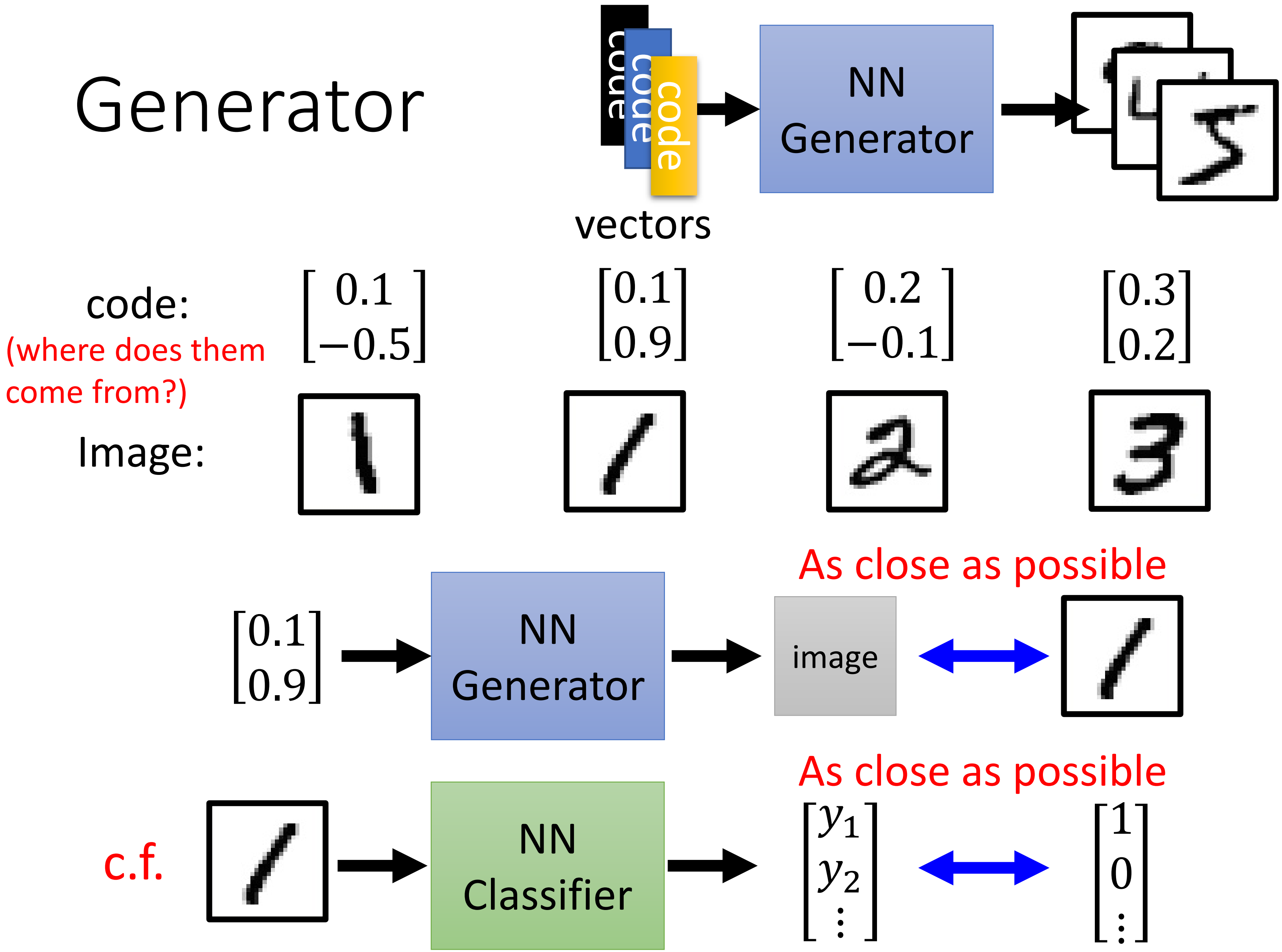
Discriminator v2

No mouth

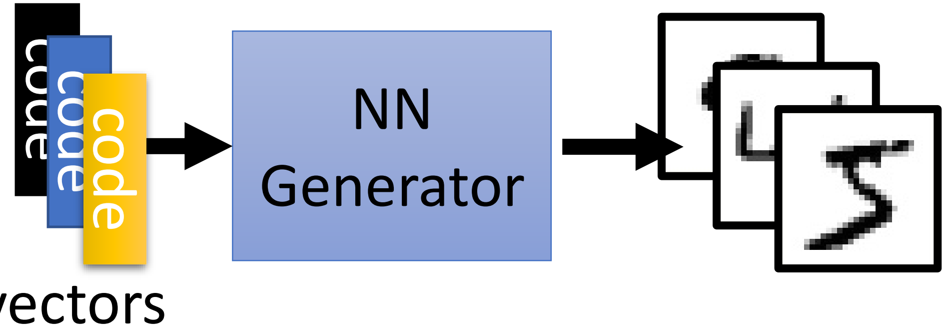
Generator v3

為什麼不自己學？

為什麼不自己做？


Generator
code
code
code
vectors
NN Generator
code:
(where does them come from?)
$\begin{bmatrix}0.1\\-0.5\end{bmatrix}$
$\begin{bmatrix}0.1\\0.9\end{bmatrix}$
$\begin{bmatrix}0.2\\-0.1\end{bmatrix}$
$\begin{bmatrix}0.3\\0.2\end{bmatrix}$
Image:
As close as possible
$\begin{bmatrix}0.1\\0.9\end{bmatrix}$
NN Generator
image
As close as possible
c.f.
NN Classifier
$\begin{bmatrix}y_1\\y_2\\\vdots\end{bmatrix}$
$\begin{bmatrix}1\\0\\\vdots\end{bmatrix}$

Generator

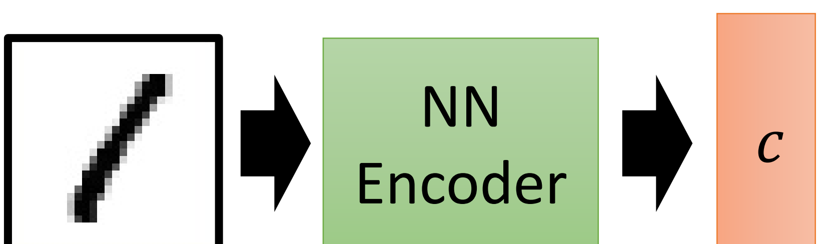


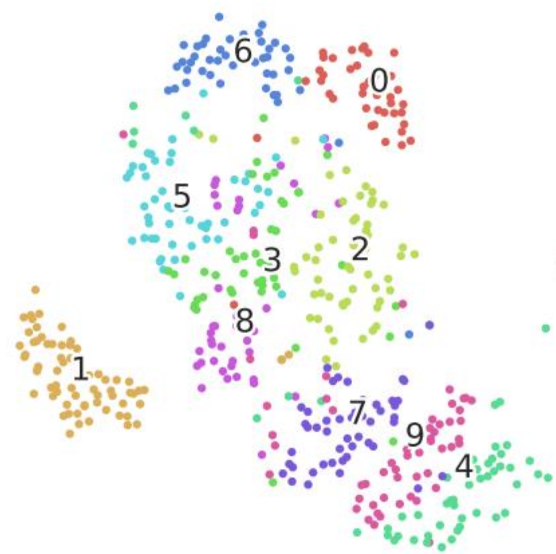
code: $\begin{bmatrix} 0.1 \\ -0.5 \end{bmatrix}$ $\begin{bmatrix} 0.1 \\ 0.9 \end{bmatrix}$ $\begin{bmatrix} 0.2 \\ -0.1 \end{bmatrix}$ $\begin{bmatrix} 0.3 \\ 0.2 \end{bmatrix}$

(where does them come from?)

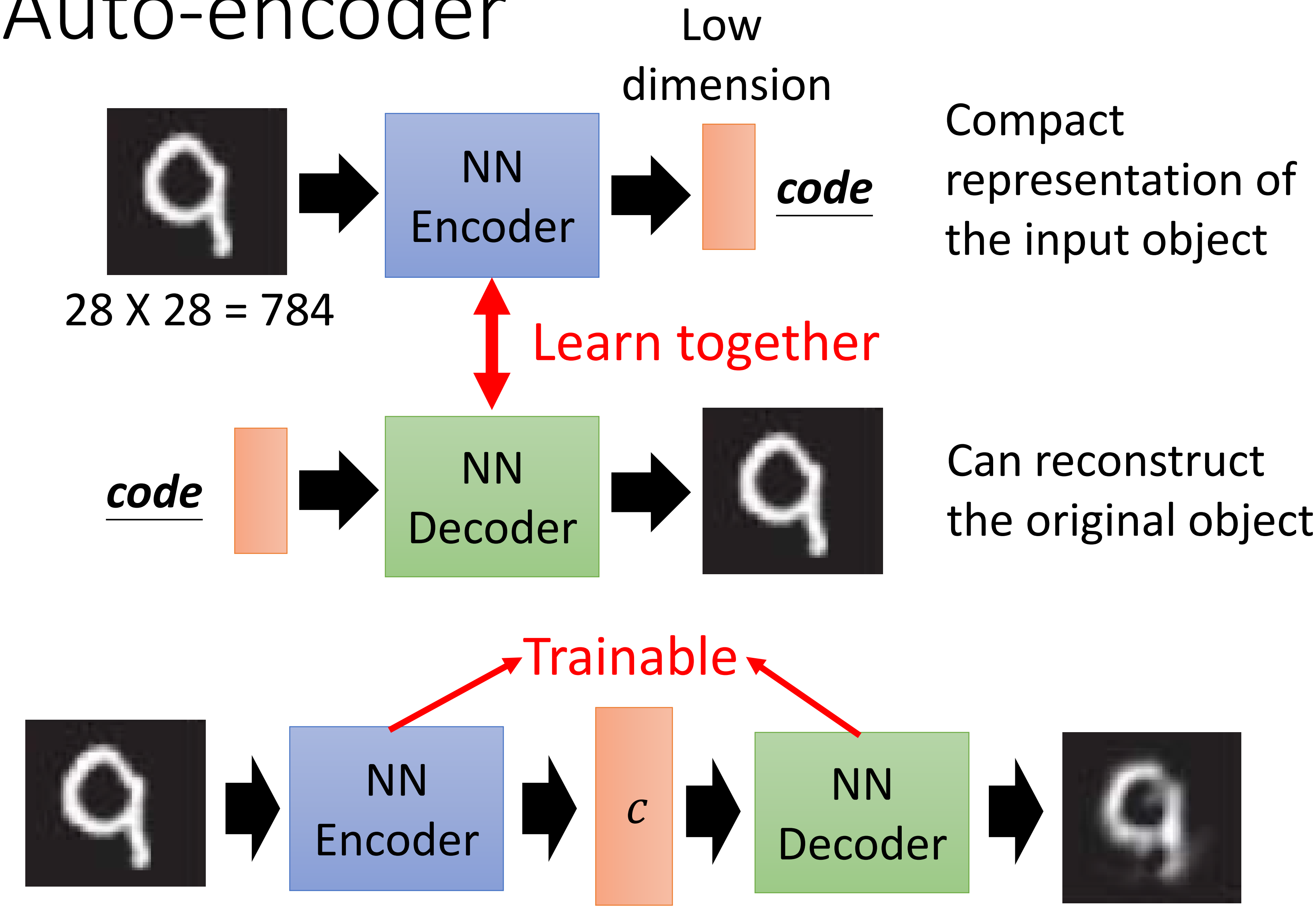
Image:

Encoder in auto-encoder provides the code ☺

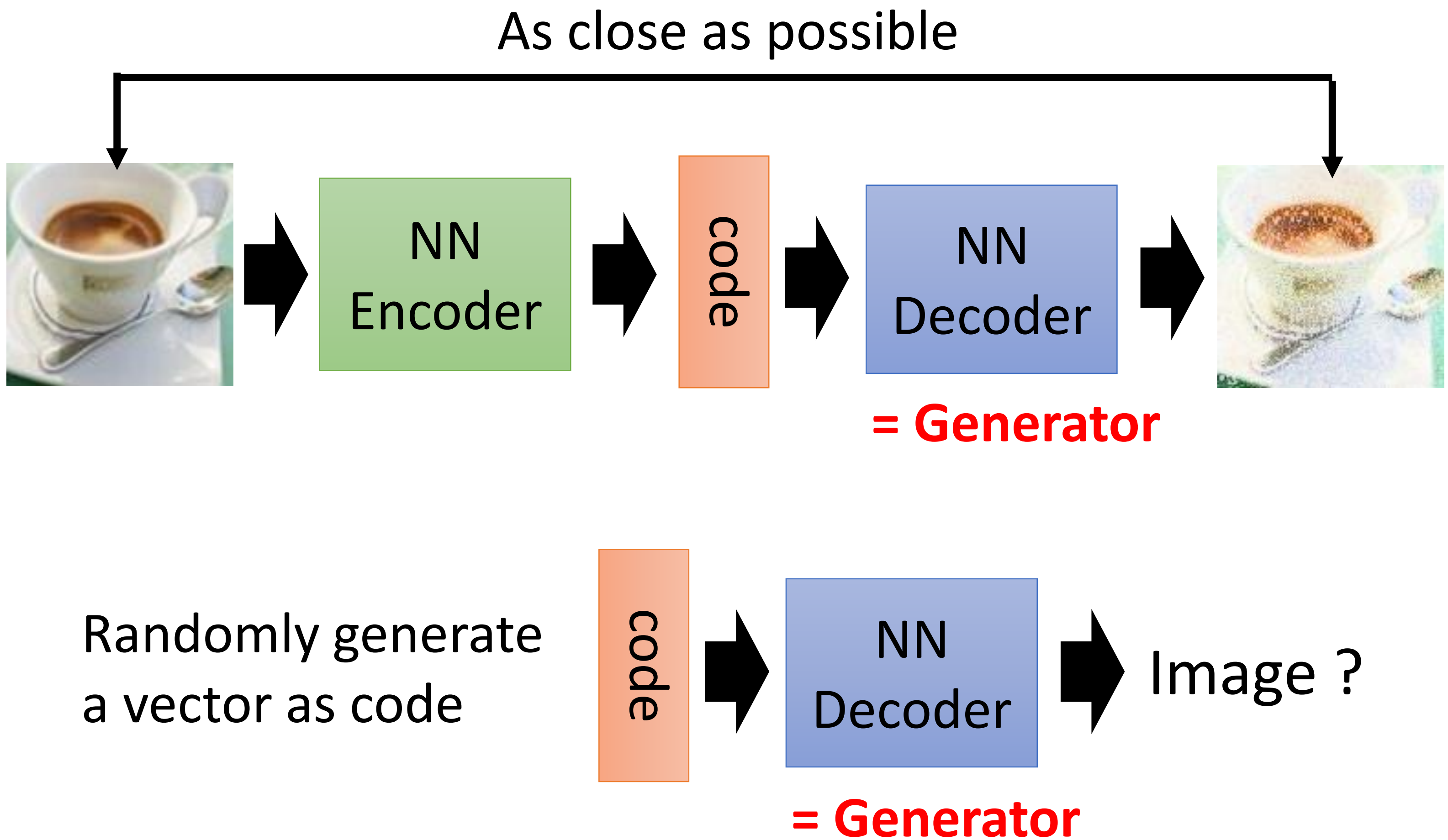




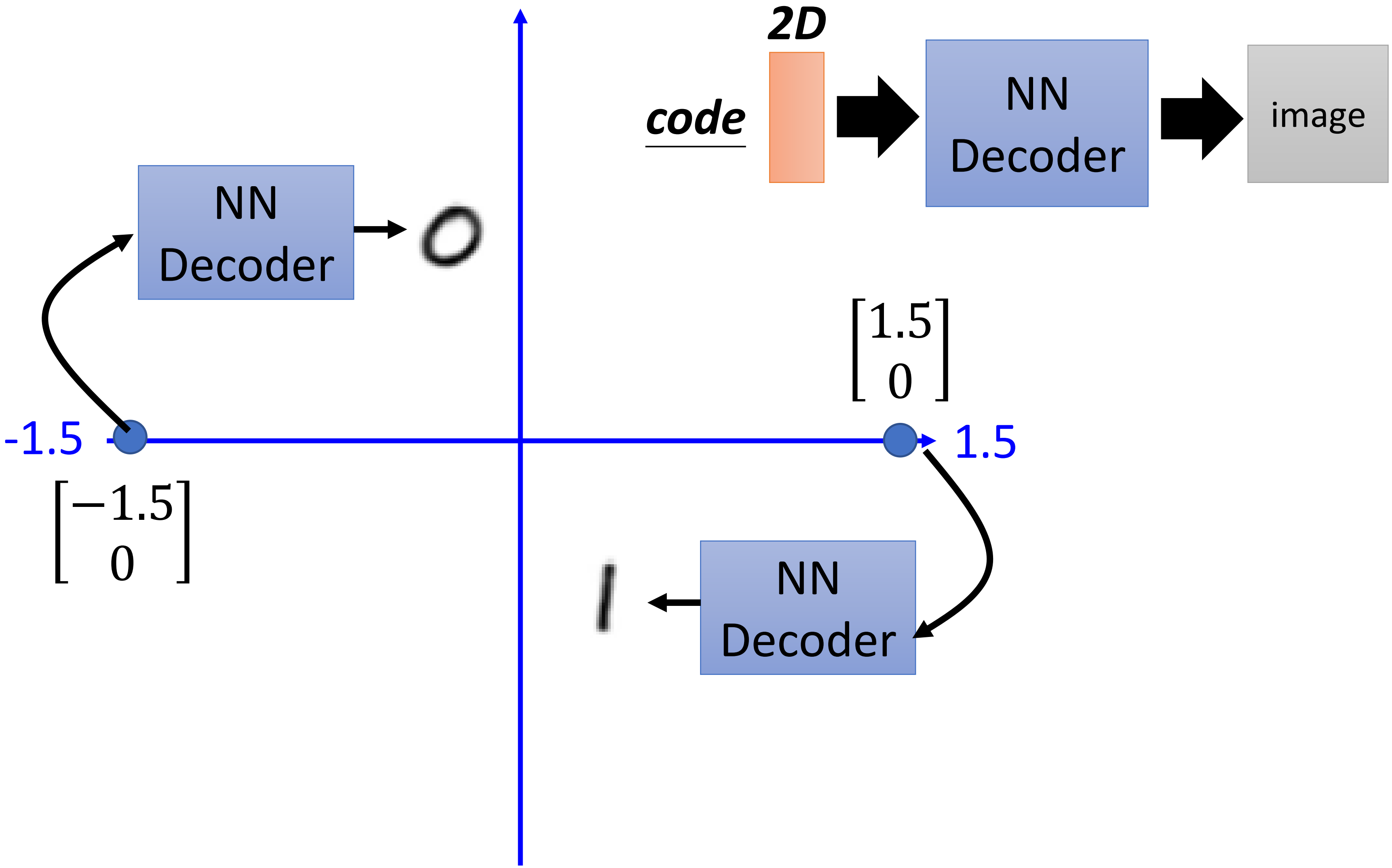
Auto-encoder



Auto-encoder

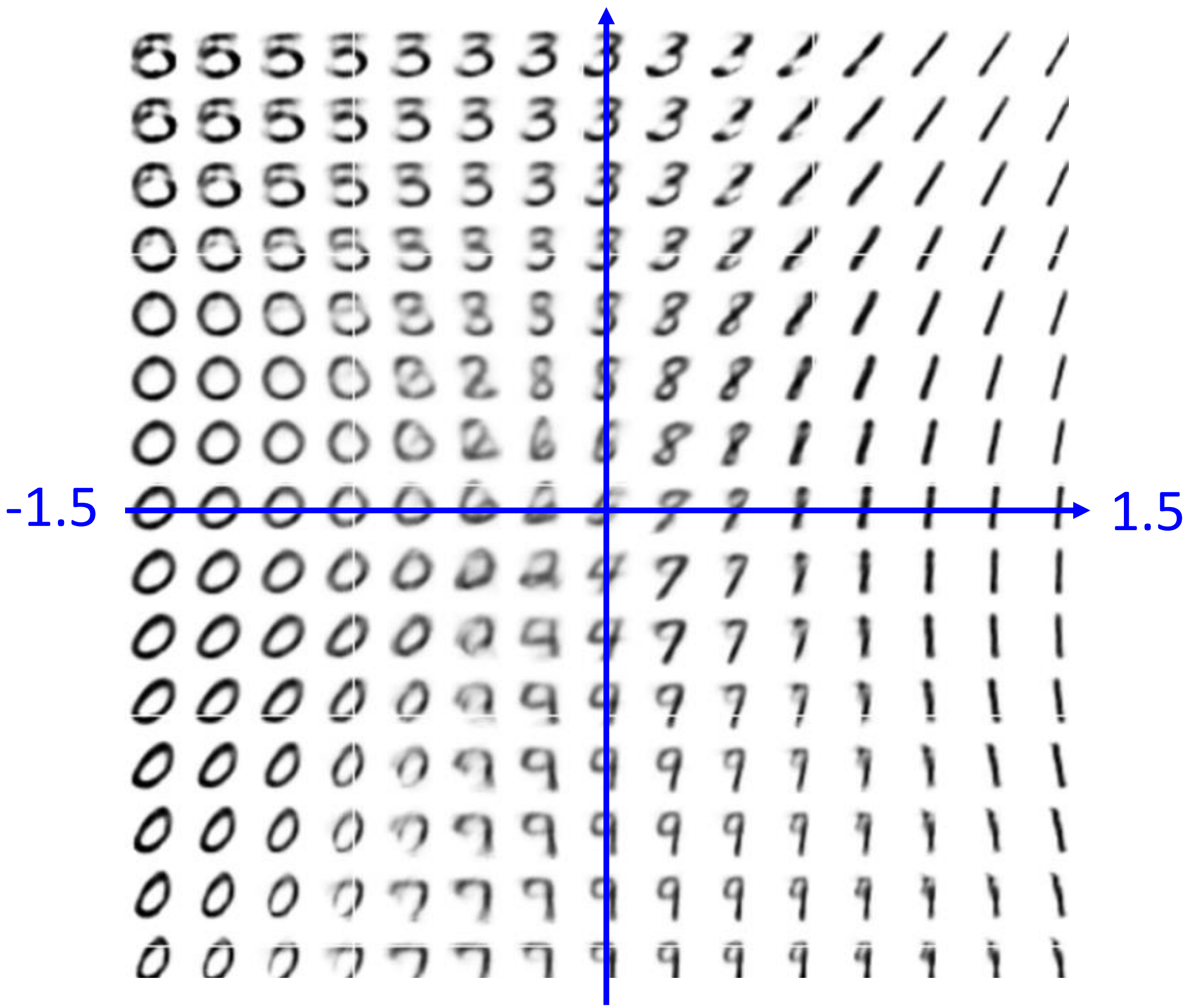


Auto-encoder

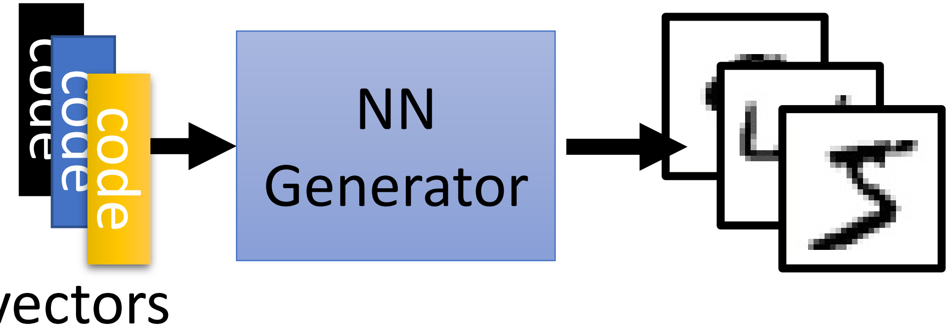


(real examples)

Auto-encoder



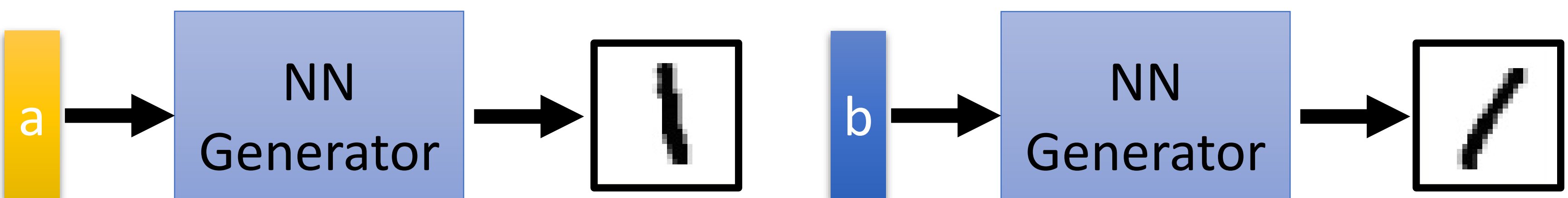
Auto-encoder

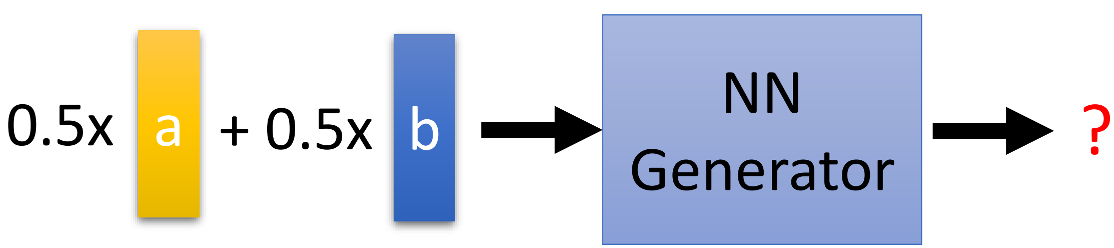


code: $\begin{bmatrix}0.1\\-0.5\end{bmatrix}$ $\begin{bmatrix}0.1\\0.9\end{bmatrix}$ $\begin{bmatrix}0.2\\-0.1\end{bmatrix}$ $\begin{bmatrix}0.3\\0.2\end{bmatrix}$

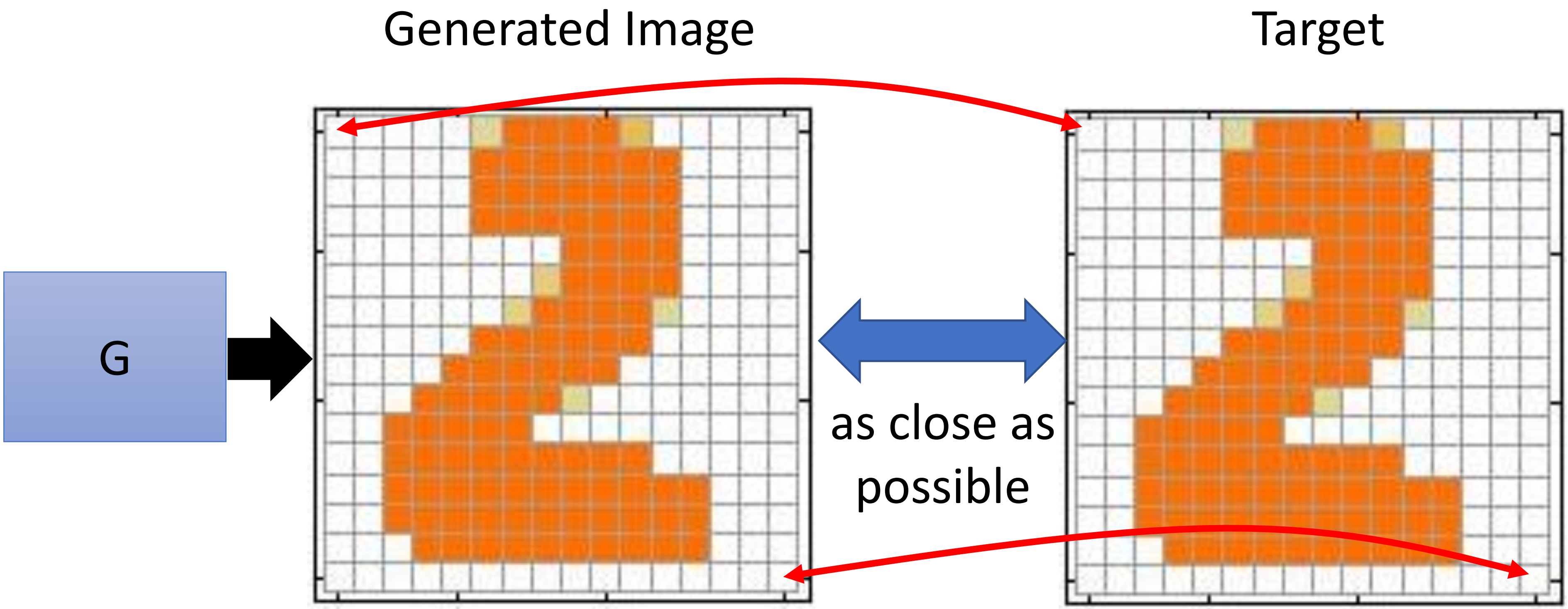
(where does them come from?)

Image:





What do we miss?



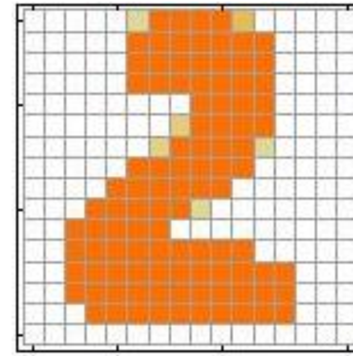
It will be fine if the generator can truly copy the target image.

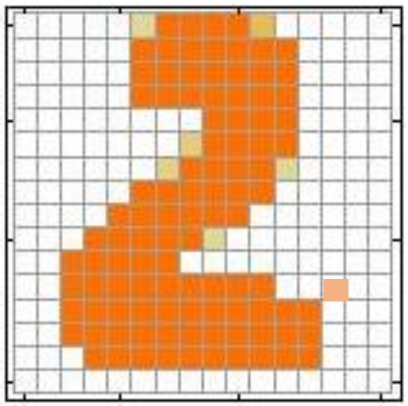
What if the generator makes some mistakes …….

Some mistakes are serious, while some are fine.

What do we miss?

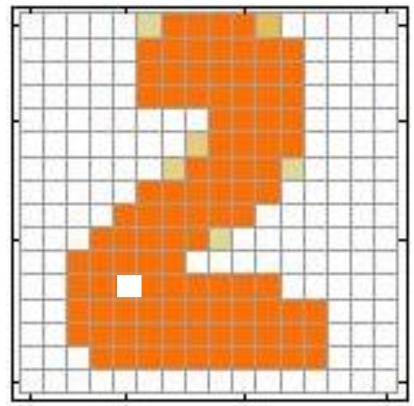
Target





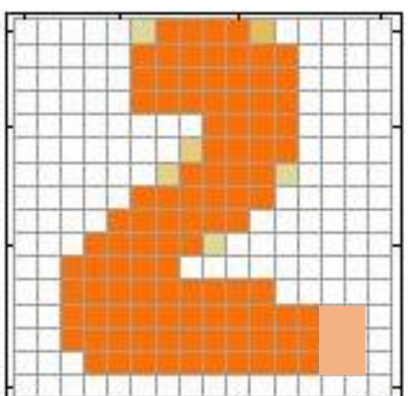
1 pixel error

我覺得不行



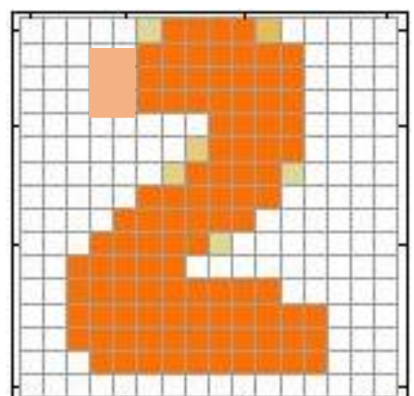
1 pixel error

我覺得不行



6 pixel errors

我覺得
其實 OK

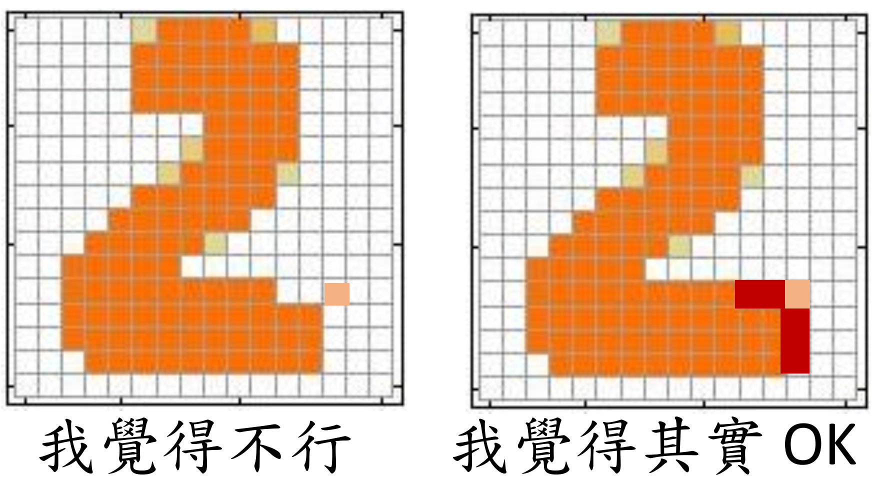


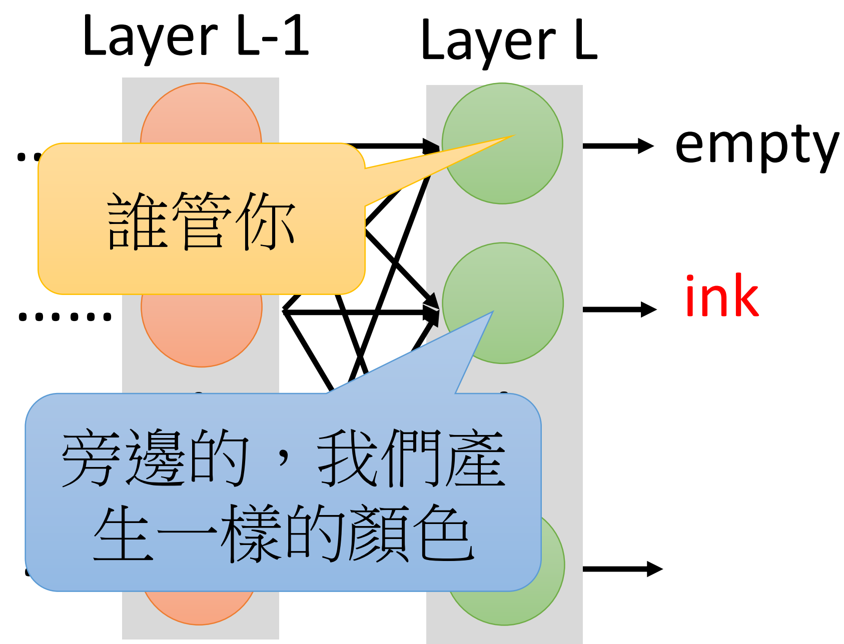
6 pixel errors

我覺得
其實 OK

What do we miss?

Each neural in output layer corresponds to a pixel.





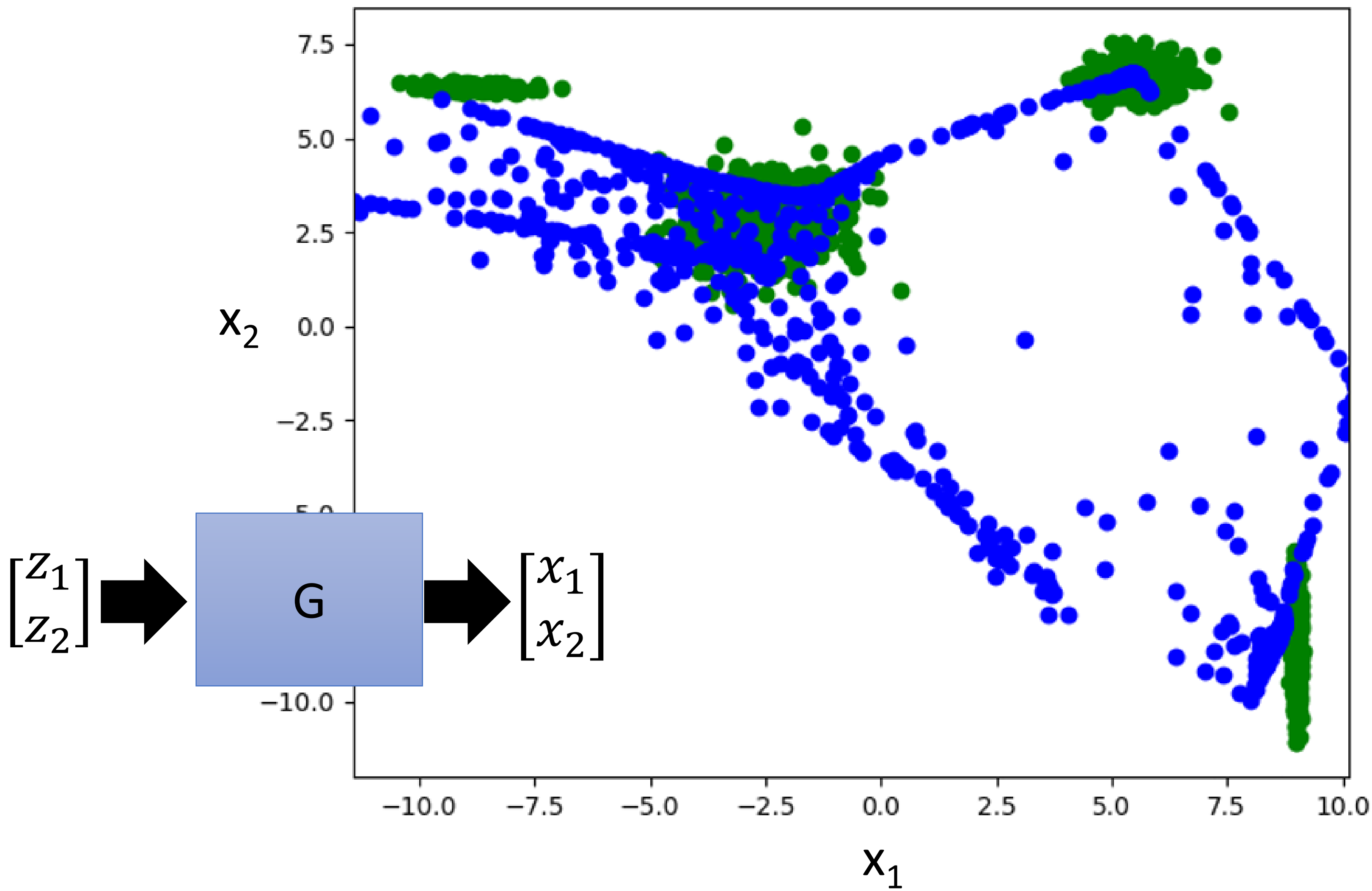
The relation between the components are critical.

The last layer generates each components independently.

Need deep structure to catch the relation between components.

感謝 黃淞楓 同學提供結果

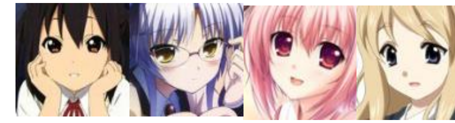
(Variational) Auto-encoder



Basic Idea of GAN (和平的比喻)

Generator (student)

Discriminator (teacher)



Generator v1

Discriminator v1

No eyes

Generator v2

Discriminator v2

No mouth

Generator v3

為什麼不自己學？

為什麼不自己做？

Discriminator

Evaluation function, Potential Function, Evaluation Function …

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

$$D : X \rightarrow R$$

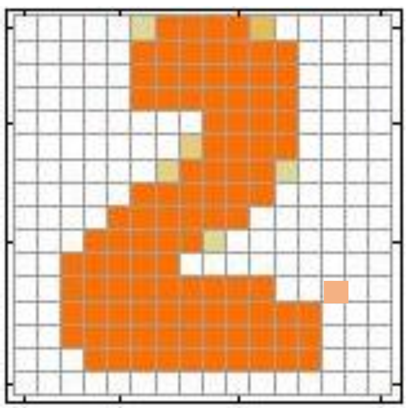
  - Input x: an object x (e.g. an image)
  - Output D(x): scalar which represents how "good" an object x is

Can we use the discriminator to generate objects?

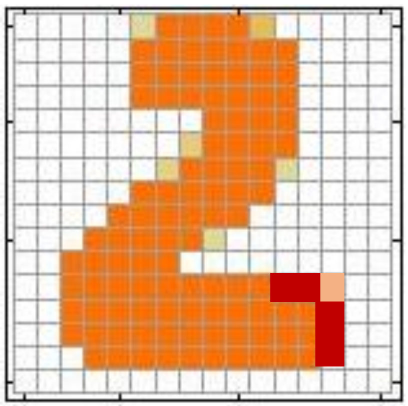
Yes.

Discriminator

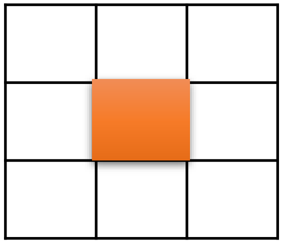
- It is easier to catch the relation between the components by top-down evaluation.



我覺得不行



我覺得其實 OK



This CNN filter is good enough.

Discriminator

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

Inference

- Generate object $\tilde{x}$ that

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

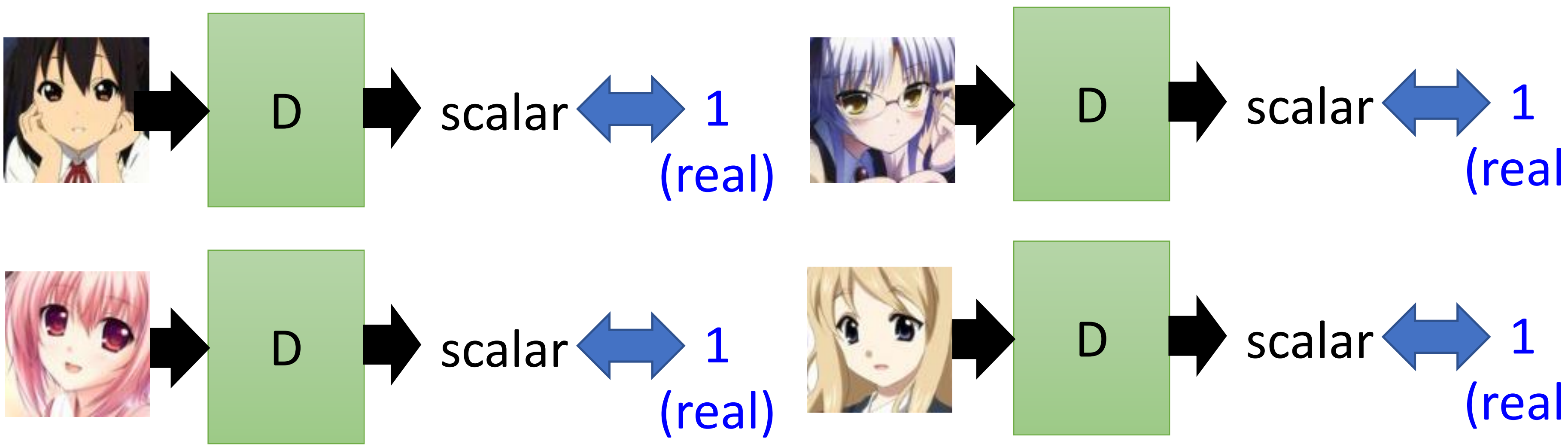
Enumerate all possible x !!!

It is feasible ???

How to learn the discriminator?

Discriminator - Training

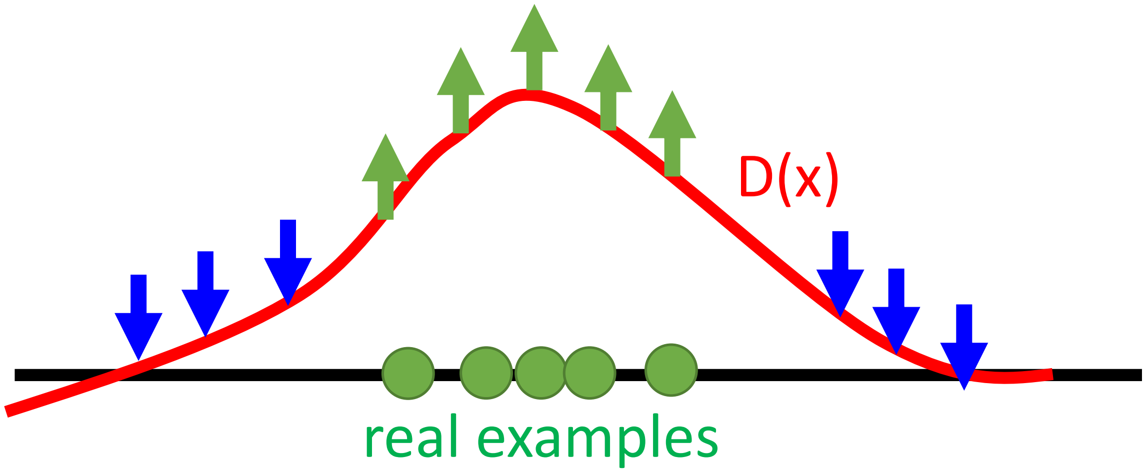
- I have some real images



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

Discriminator training needs some negative examples.

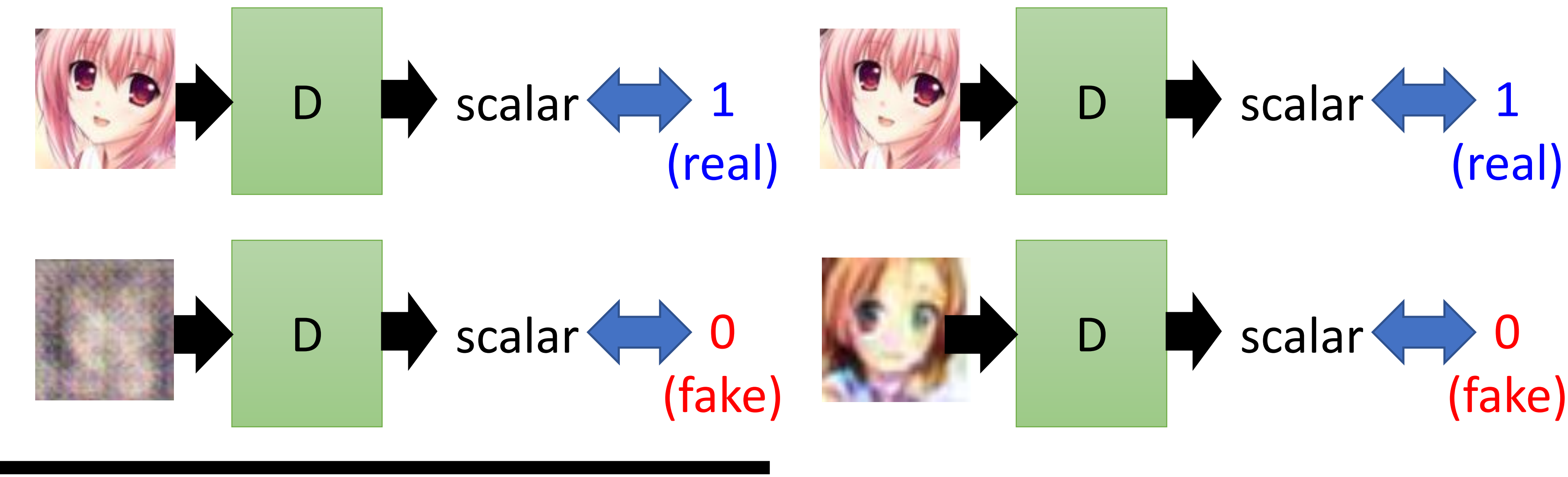
Discriminator - Training

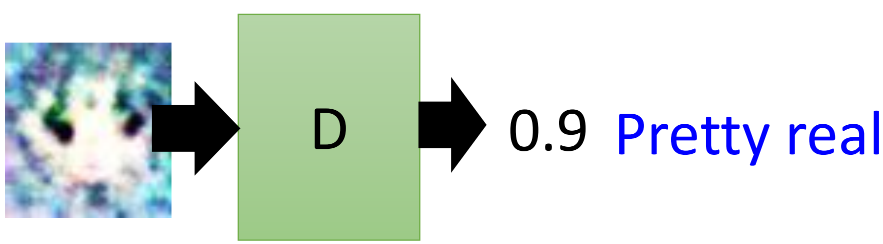


In practice, you cannot decrease all the x other than real examples.

Discriminator - Training

- Negative examples are critical.



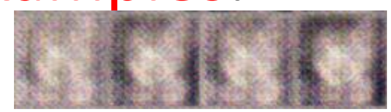


How to generate realistic negative examples?

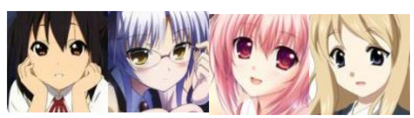
Discriminator - Training

- **General Algorithm**

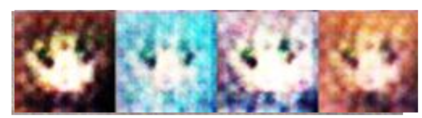
  - Given a set of positive examples, randomly generate a set of negative examples.

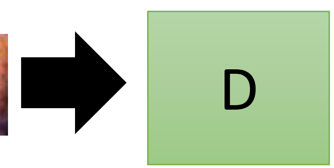


  - In each iteration
    - Learn a discriminator D that can discriminate positive and negative examples.



v.s.

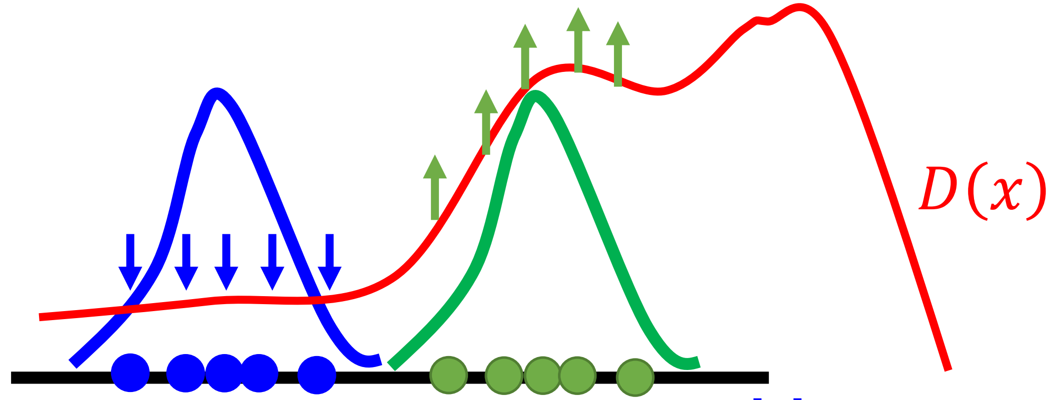




    - Generate negative examples by discriminator D

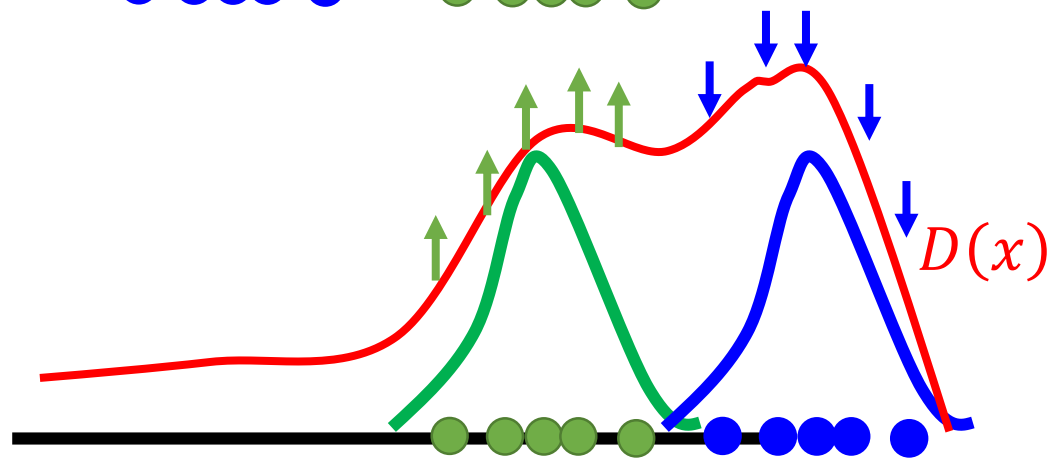
$$\tilde{x} = \arg\max_{x \in X} D(x)$$

Discriminator - Training

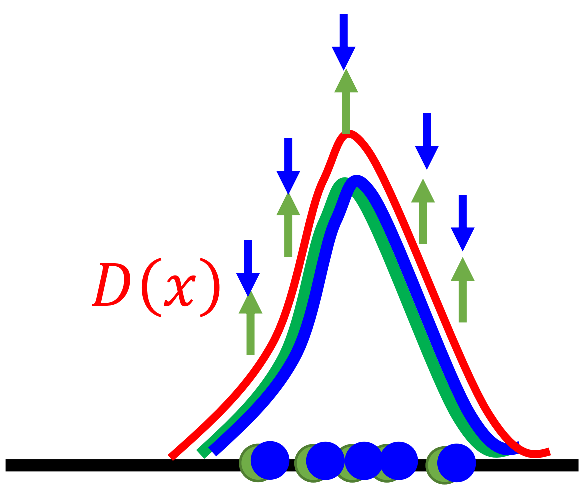


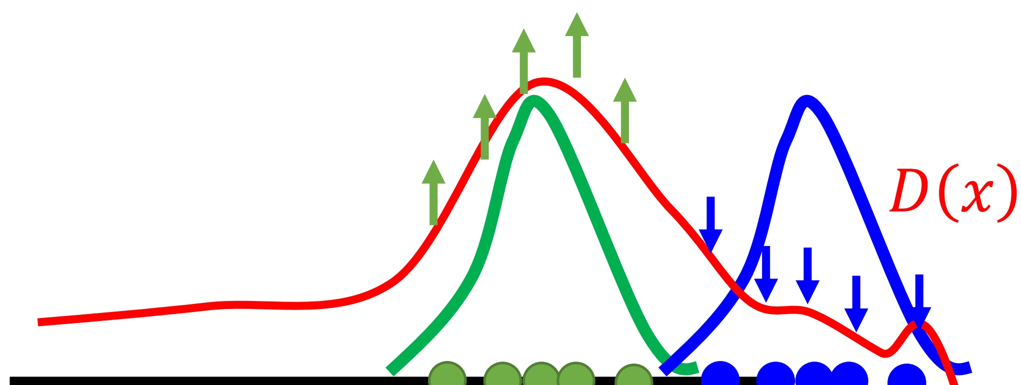
real

generated



In the end ......





Structured Learning

- Structured Perceptron
- Structured SVM
- Gibbs sampling
- Hidden information
- Application: sequence labelling, summarization

Graphical Model

Bayesian Network (Directed Graph)

Markov Random Field (Undirected Graph)

Energy-based Model: http://www.cs.nyu.edu/~yann/research/ebm/

Conditional Random Field

Markov Logic Network

Boltzmann Machine

Segmental CRF

(Only list some of the approaches)

Restricted Boltzmann Machine

Generator v.s. Discriminator

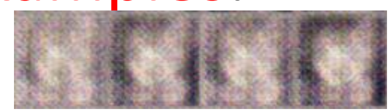
- ***Generator***
- Pros:
  - Easy to generate even with deep model
- Cons:
  - Imitate the appearance
  - Hard to learn the correlation between components

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

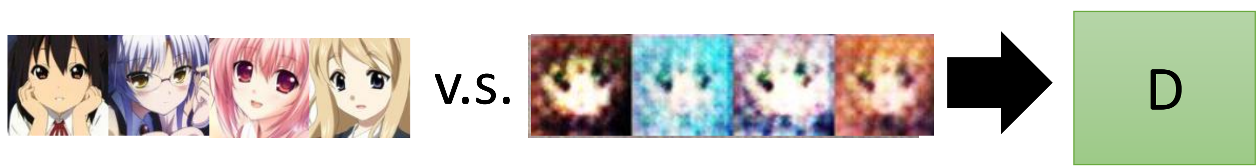
Generator + Discriminator

- **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.

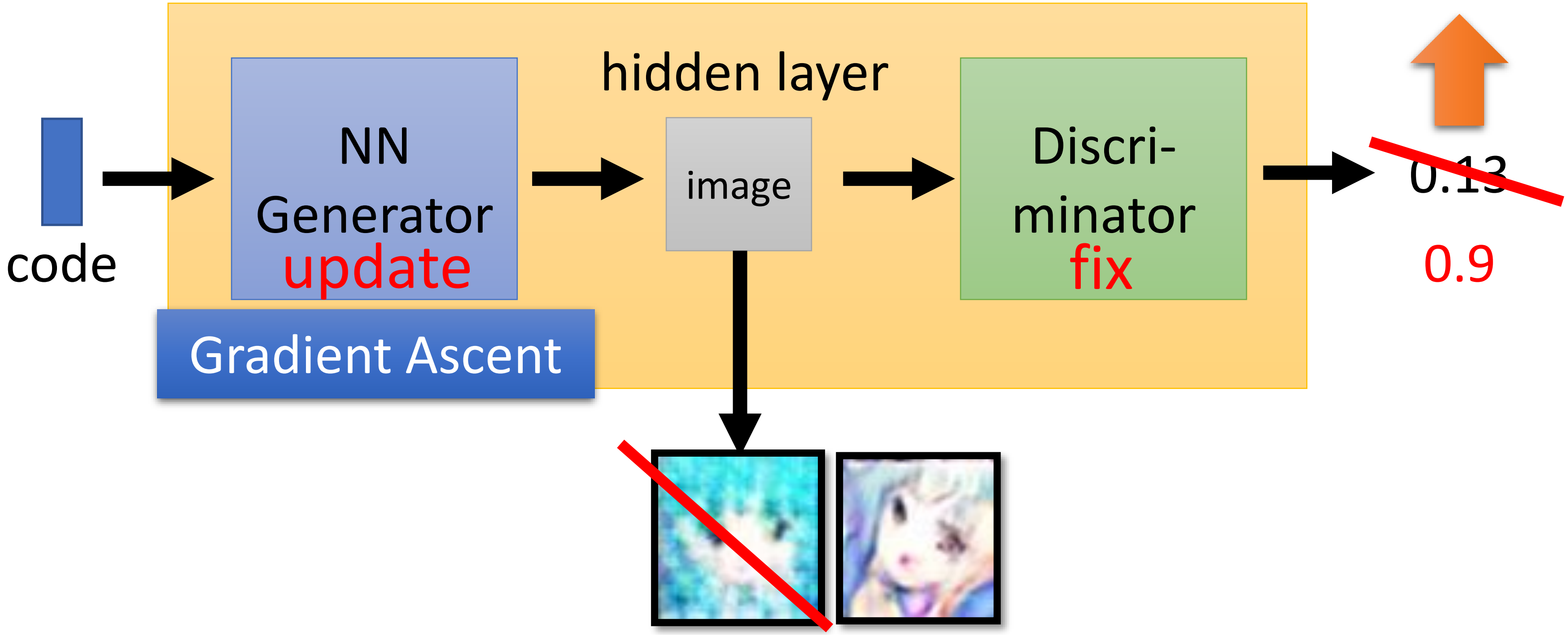


    - Generate negative examples by discriminator D

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

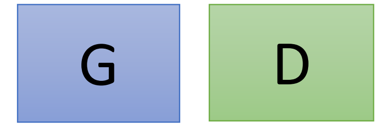
Generating Negative Examples

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

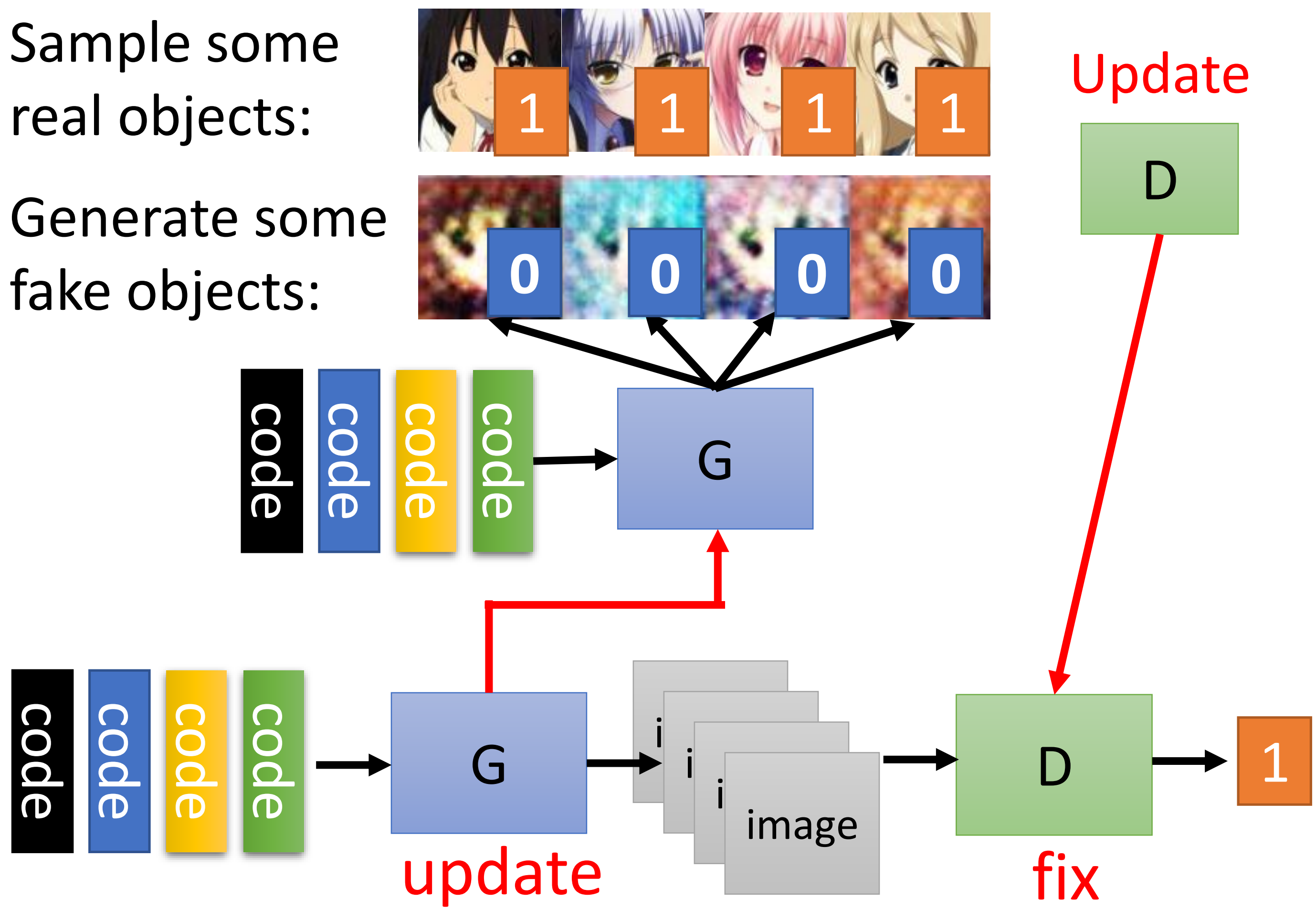


Algorithm

- Initialize generator and discriminator

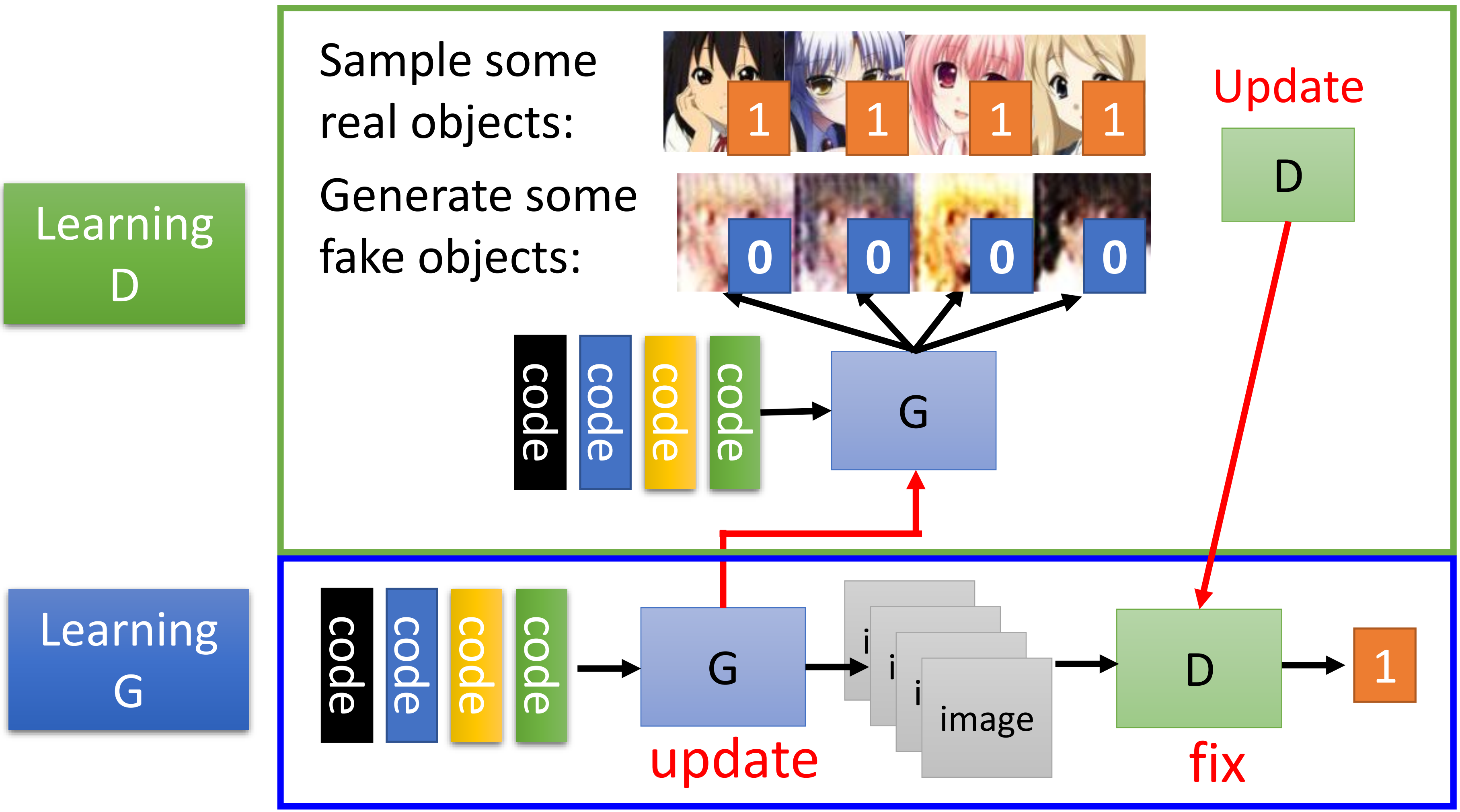


- In each training iteration:



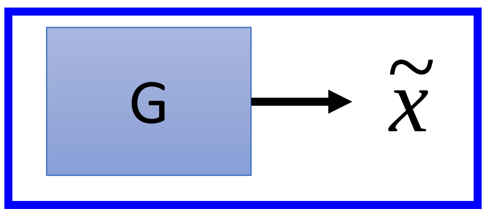
Algorithm

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



Benefit of GAN

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



=

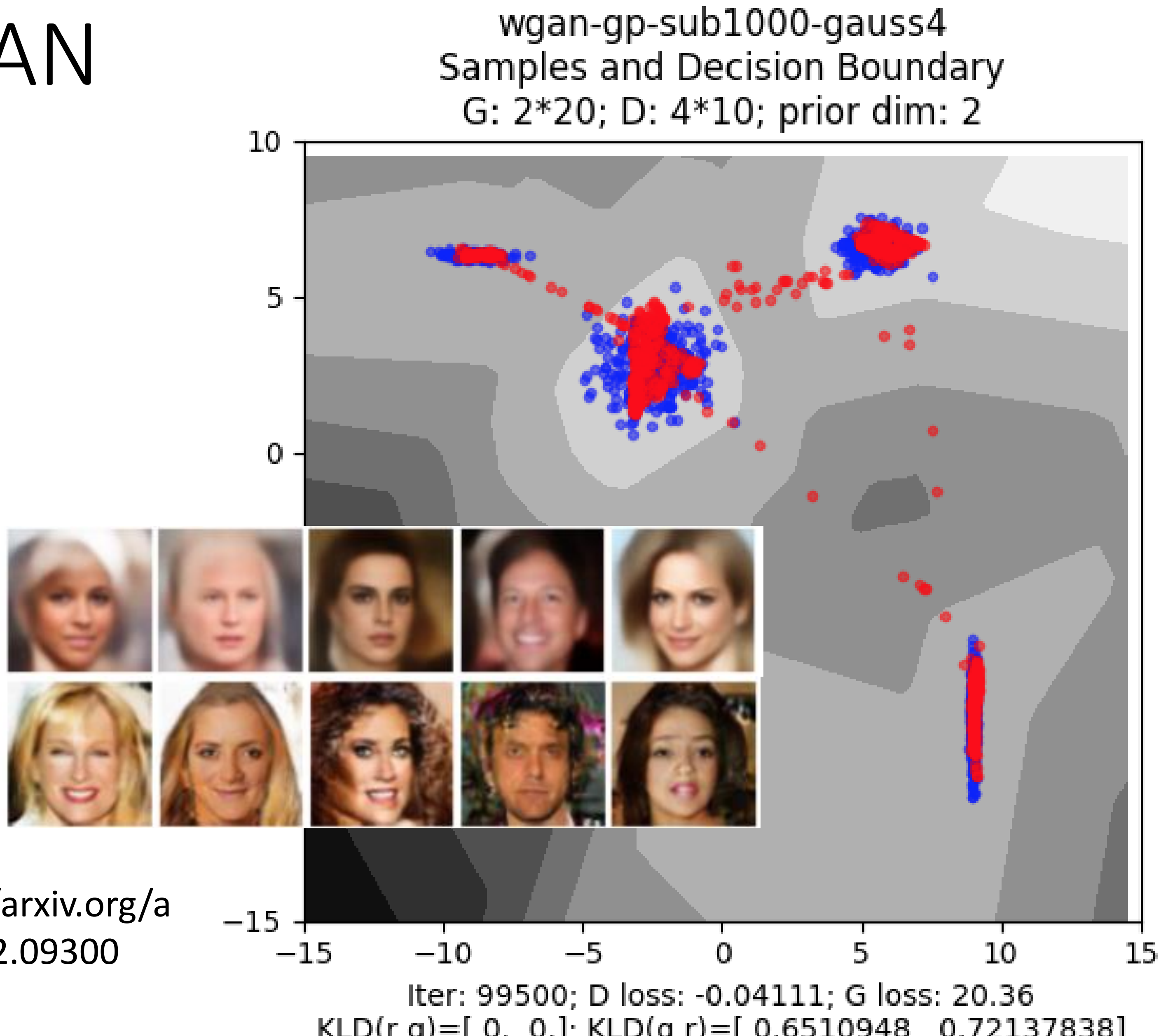
$$\boxed{\tilde{x} = \arg\max_{x \in X} D(x)}$$

efficient

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

感謝 段逸林 同學提供結果

GAN



VAE

GAN

https://arxiv.org/abs/1512.09300

Outline

- Basic Idea of GAN
- When do we need GAN?
- GAN as structured learning algorithm
- Conditional Generation by GAN
  - Modifying input code
  - Paired data
  - Unpaired data
  - Application: Intelligent Photoshop

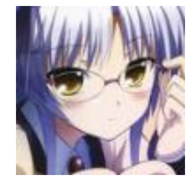
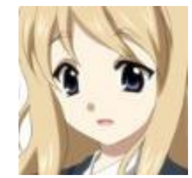
Conditional Generation

Generation

$$\begin{bmatrix} 0.3 \\ -0.1 \\ \vdots \\ -0.7 \end{bmatrix} \begin{bmatrix} 0.1 \\ -0.1 \\ \vdots \\ 0.7 \end{bmatrix} \begin{bmatrix} -0.3 \\ 0.1 \\ \vdots \\ 0.9 \end{bmatrix}$$

In a specific range

NN Generator

Conditional Generation

"Girl with red hair and red eyes"

"Girl with yellow ribbon"

NN Generator


Outline

Basic Idea of GAN

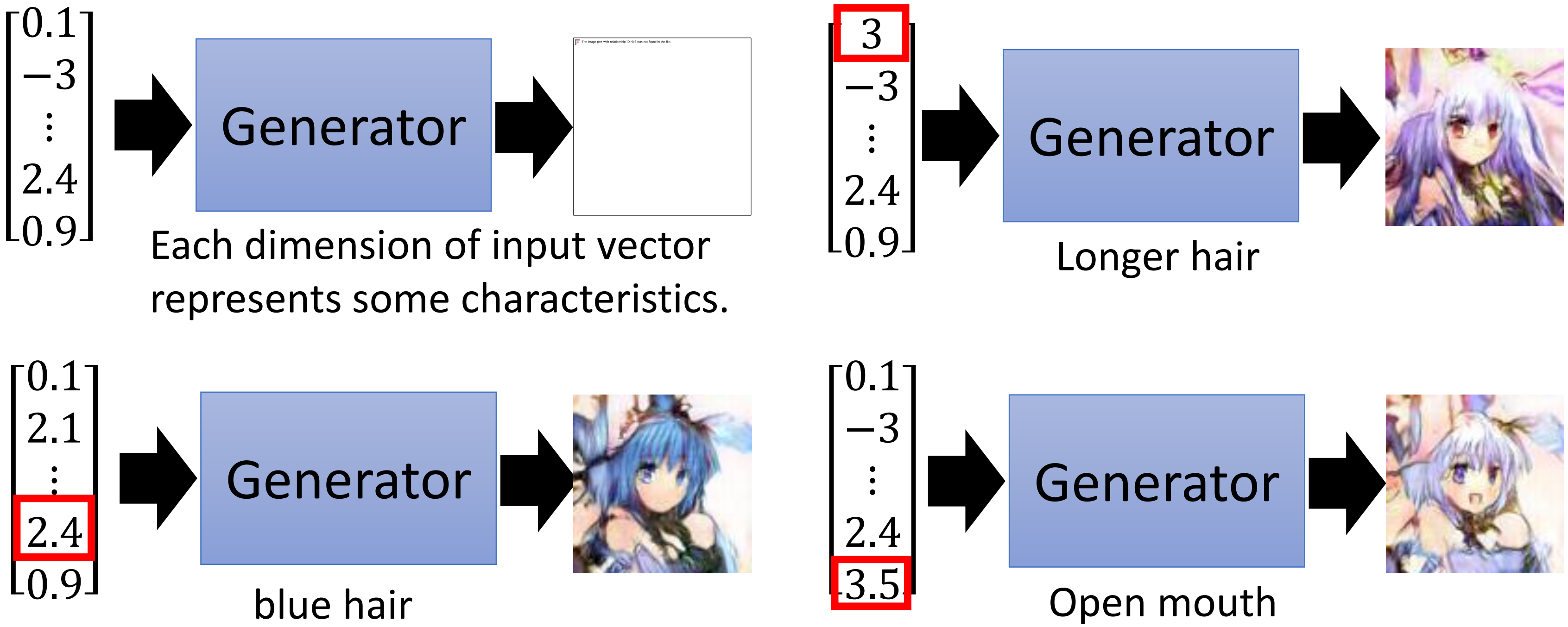
When do we need GAN?

GAN as structured learning algorithm

Conditional Generation by GAN

- Modifying input code
- Paired data
- Unpaired data
- Application: Intelligent Photoshop

Modifying Input Code



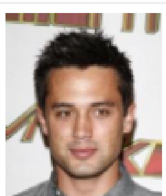
- ➢The input code determines the generator output.
- ➢Understand the meaning of each dimension to control the output.

Connecting Code and Attribute

(c) Hair style

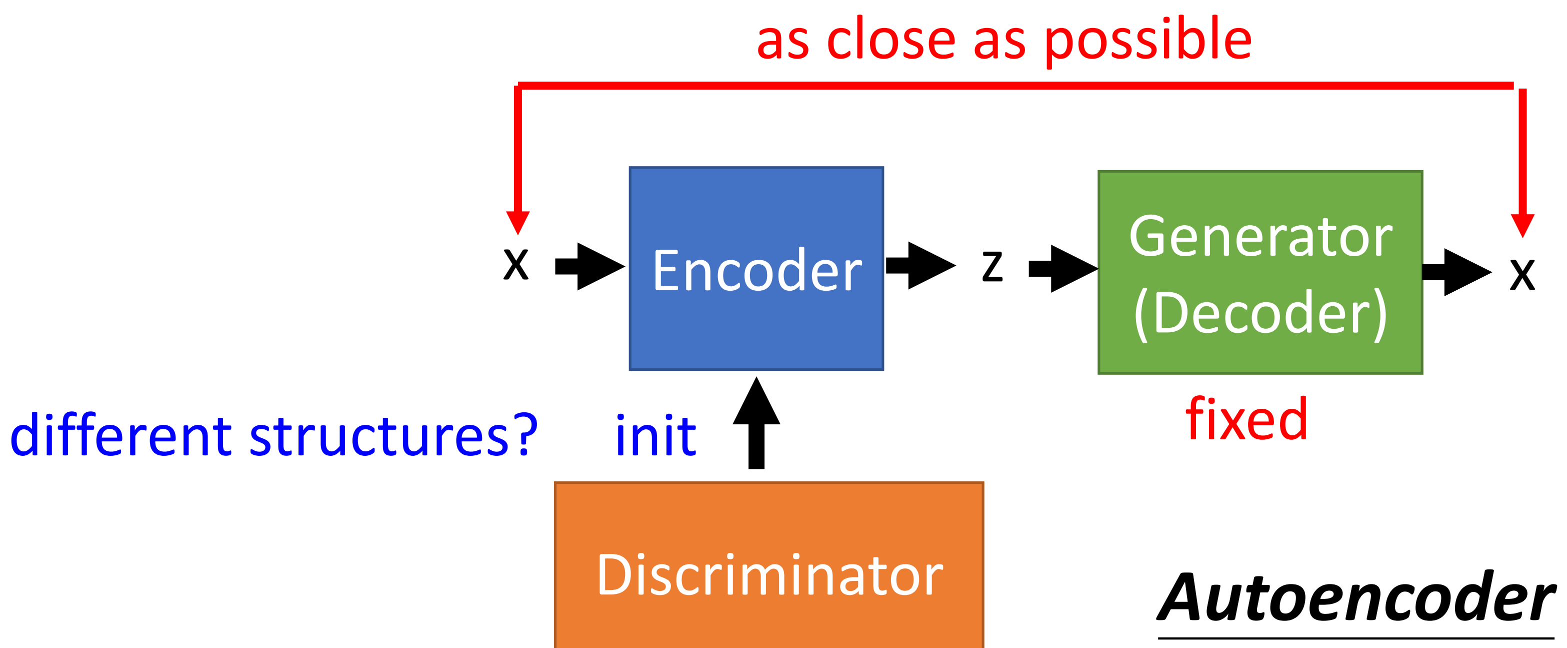
(d) Emotion

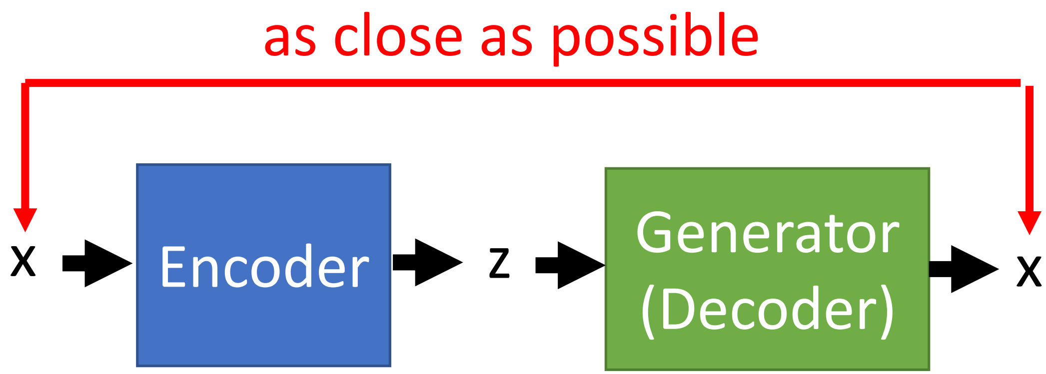
CelebA

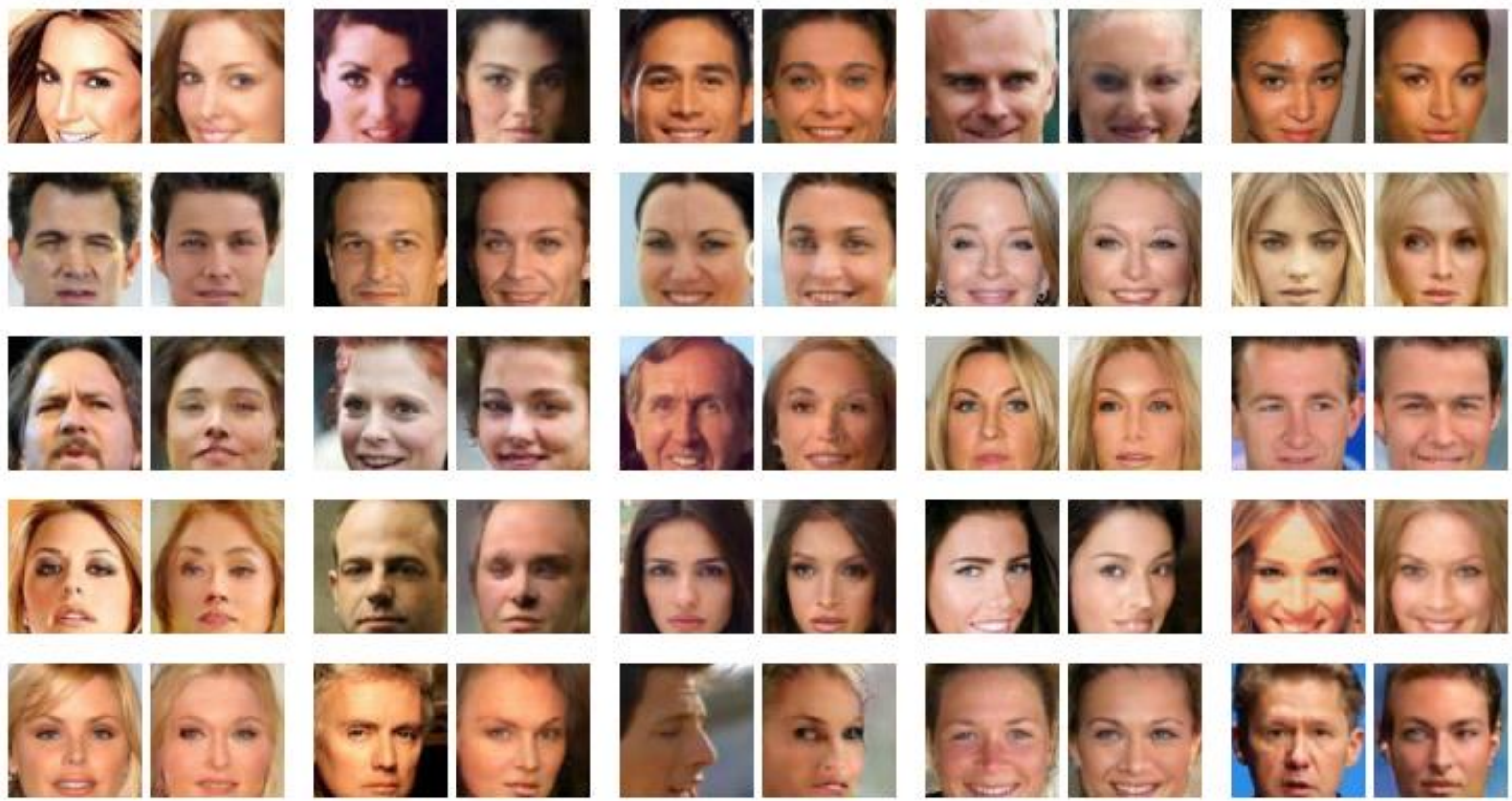
Image	Attributes
	Arched eyebrows, attractive, brown hair, heavy makeup, high cheekbones, mouth slightly open, no beard, pointy nose, smiling, straight hair, wearing earrings, wearing lipstick, young.
	5 o'clock shadows, attractive, bags under eyes, big lips, big nose, black hair, bushy eyebrows, male, no beard, pointy nose, straight hair, young.

GAN+Autoencoder

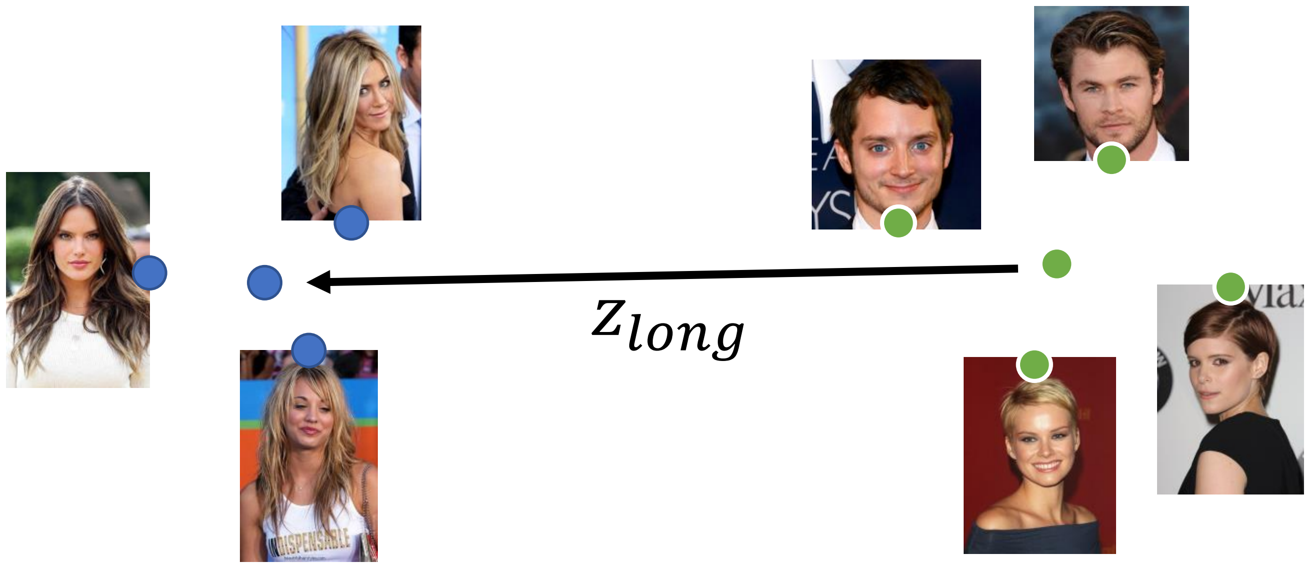
- We have a generator (input z, output x)
- However, given x, how can we find z?
    - Learn an encoder (input x, output z)




as close as possible
x
Encoder
z
Generator
(Decoder)
x



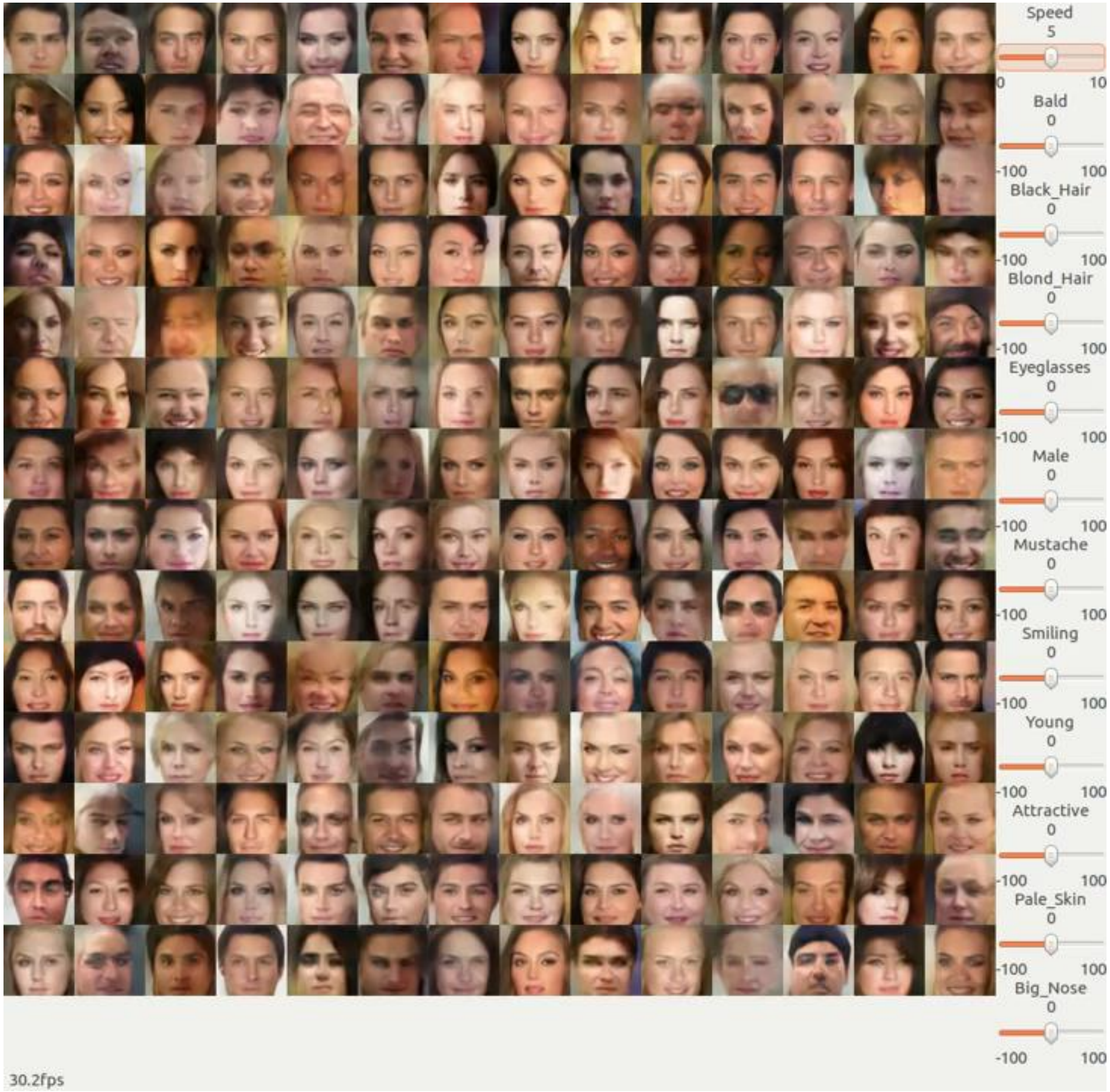
Attribute Representation



$$z_{long} = \frac{1}{N_1} \sum_{x \in long} En(x) - \frac{1}{N_2} \sum_{x' \notin long} En(x')$$

Short Hair $\quad x \Rightarrow En(x) + z_{long} = z' \Rightarrow Gen(z') \quad$ Long Hair

Photo Editing



https://www.youtube.com/watch?v=kPEIJJsQr7U

Outline

Basic Idea of GAN

When do we need GAN?

GAN as structured learning algorithm

Conditional Generation by GAN

- Modifying input code
- Paired data
- Unpaired data
- Application: Intelligent Photoshop

Conditional GAN

c^1: a dog is running $\hat{x}^1$: 

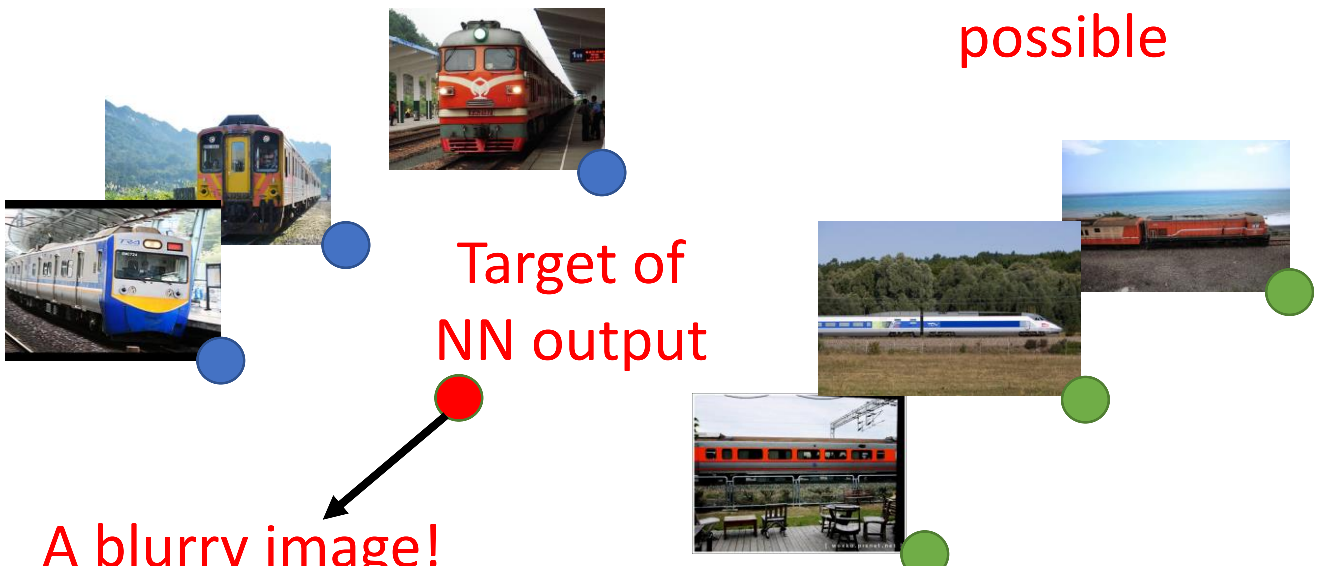
c^2: a bird is flying $\hat{x}^2$: 

- ***Text to image*** by traditional supervised learning

c^1: a dog is running → NN → Image ↔ $\hat{x}^1$: 

as close as possible

Text: “train”



Conditional GAN

c: train

Prior distribution z

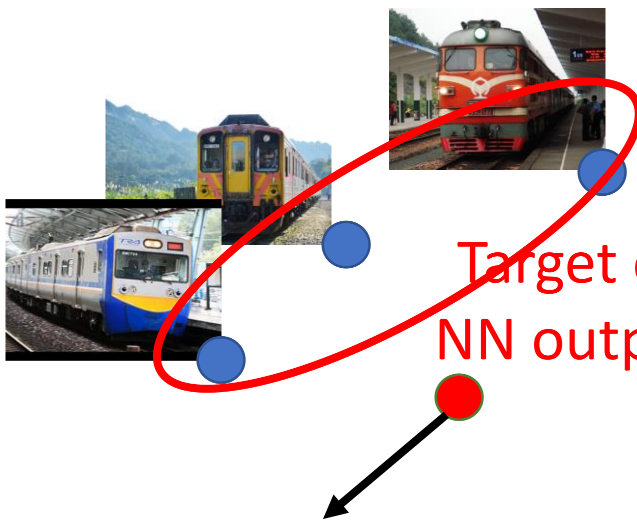
G

Image $x = G(c,z)$

It is a distribution.

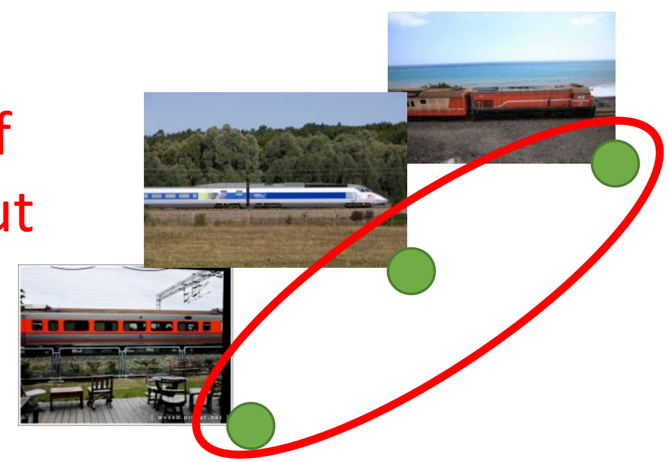
Approximate the distribution below

Text: "train"

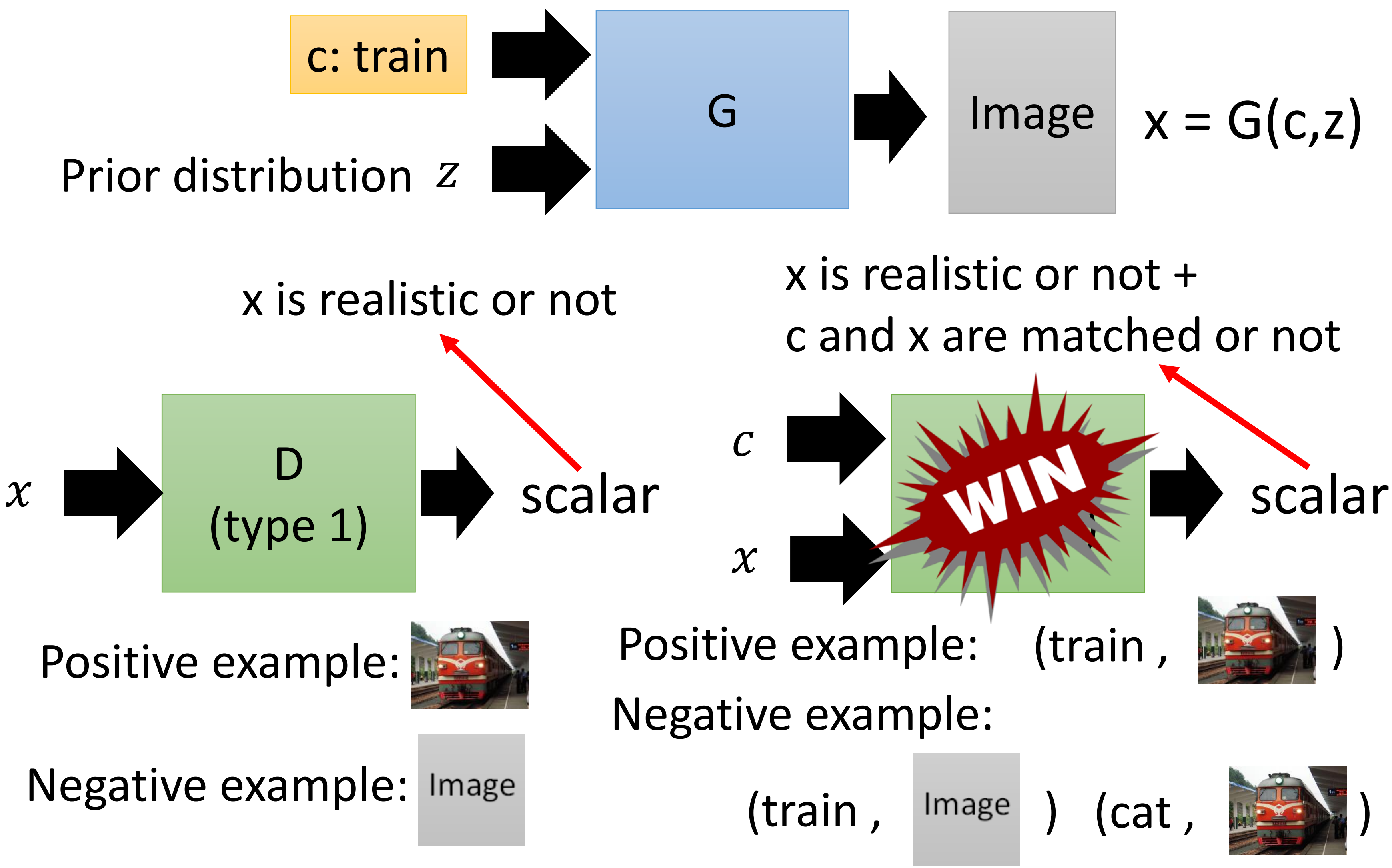


Target of NN output

A blurry image!



Conditional GAN



Text to Image - Results

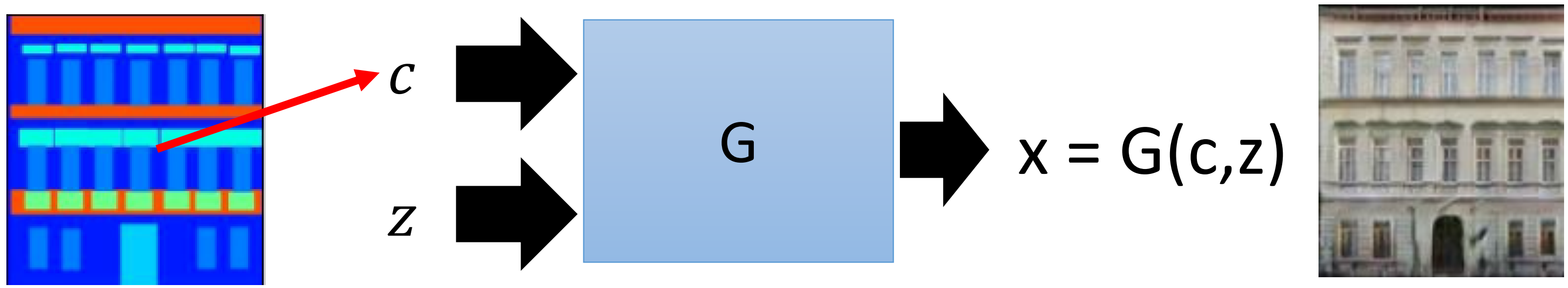
"red flower with black center"

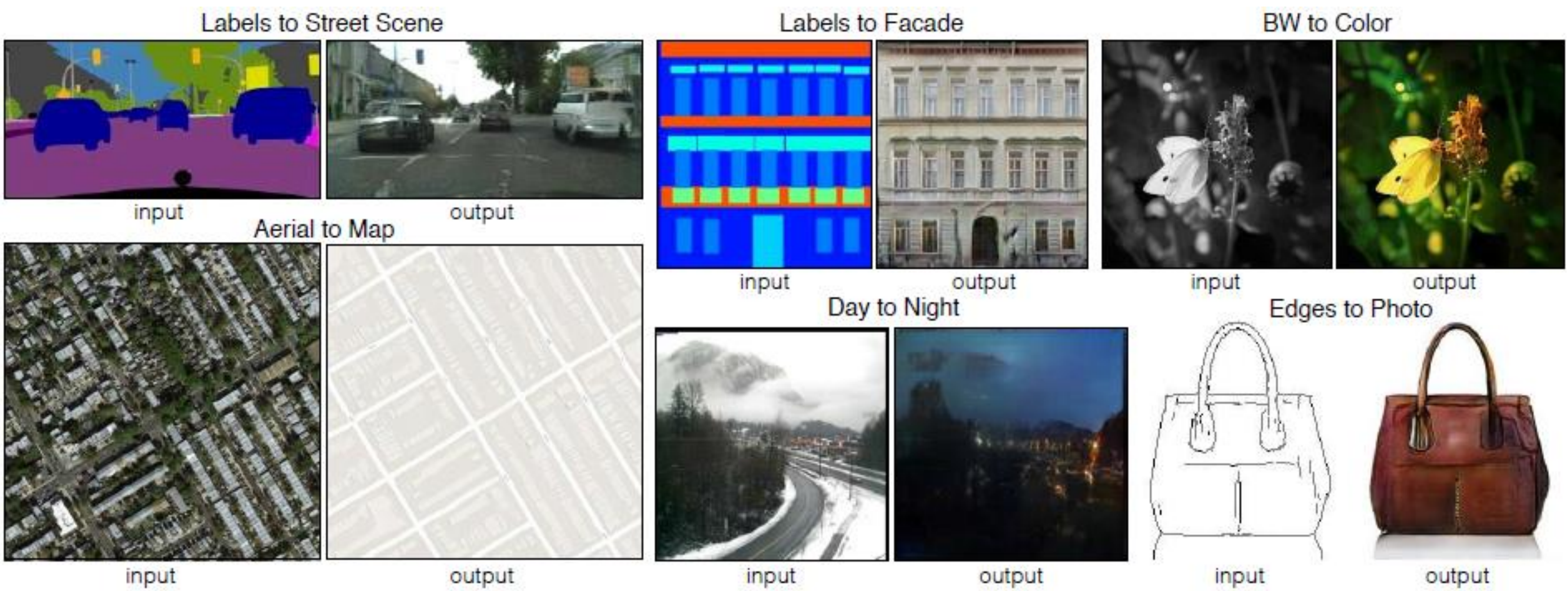
Caption	Image
this flower has white petals and a yellow stamen	
the center is yellow surrounded by wavy dark purple petals	
this flower has lots of small round pink petals	

Text to Image
- Results

Caption	Image
a pitcher is about to throw the ball to the batter	
a group of people on skis stand in the snow	
a man in a wet suit riding a surfboard on a wave	

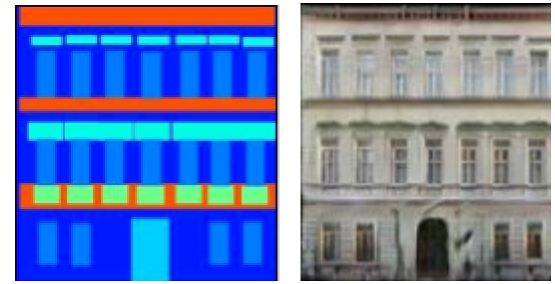
Image-to-image



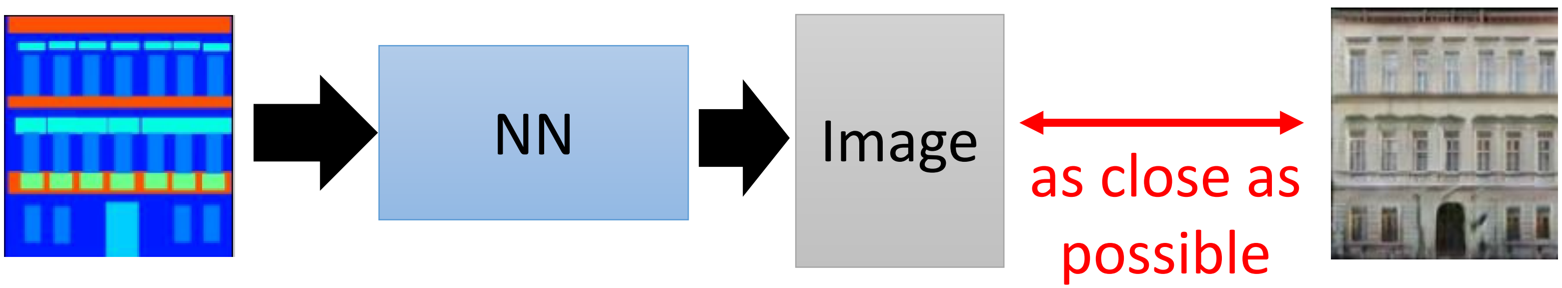


https://arxiv.org/pdf/1611.07004

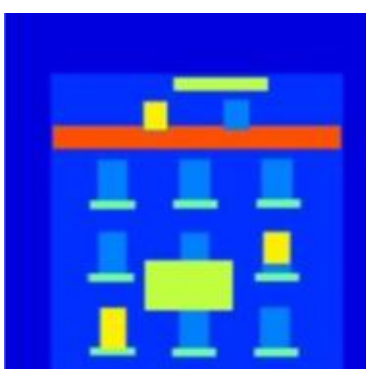
Image-to-image

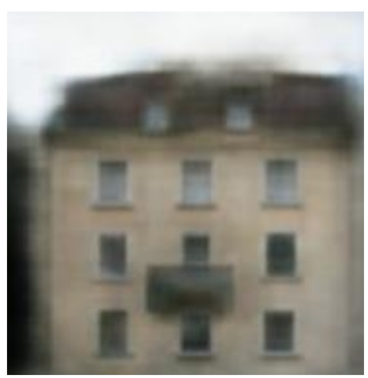


- Traditional supervised approach



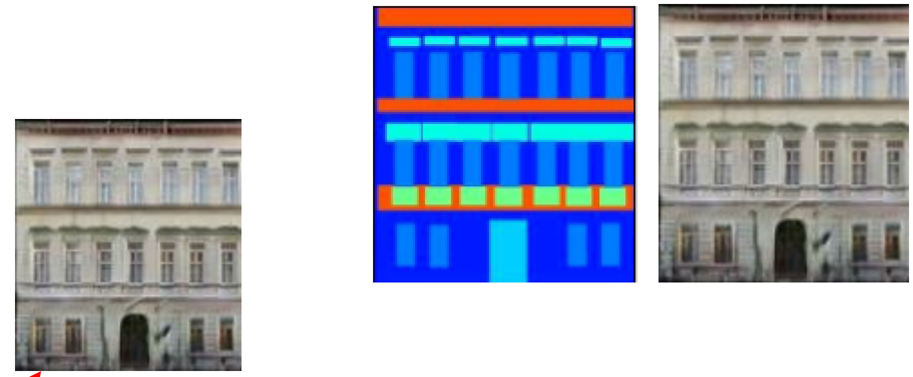
Testing:


input

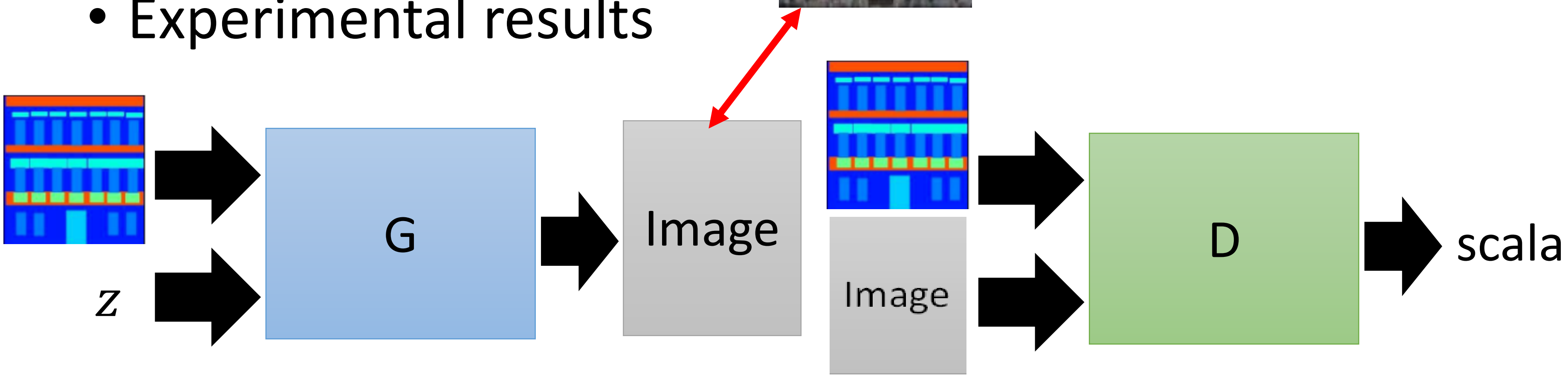

close

It is blurry because it is the average of several images.

Image-to-image



- Experimental results



Testing:

input   close   GAN   GAN + close

Image super resolution

- Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, Wenzhe Shi, “Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network”, CVPR, 2016

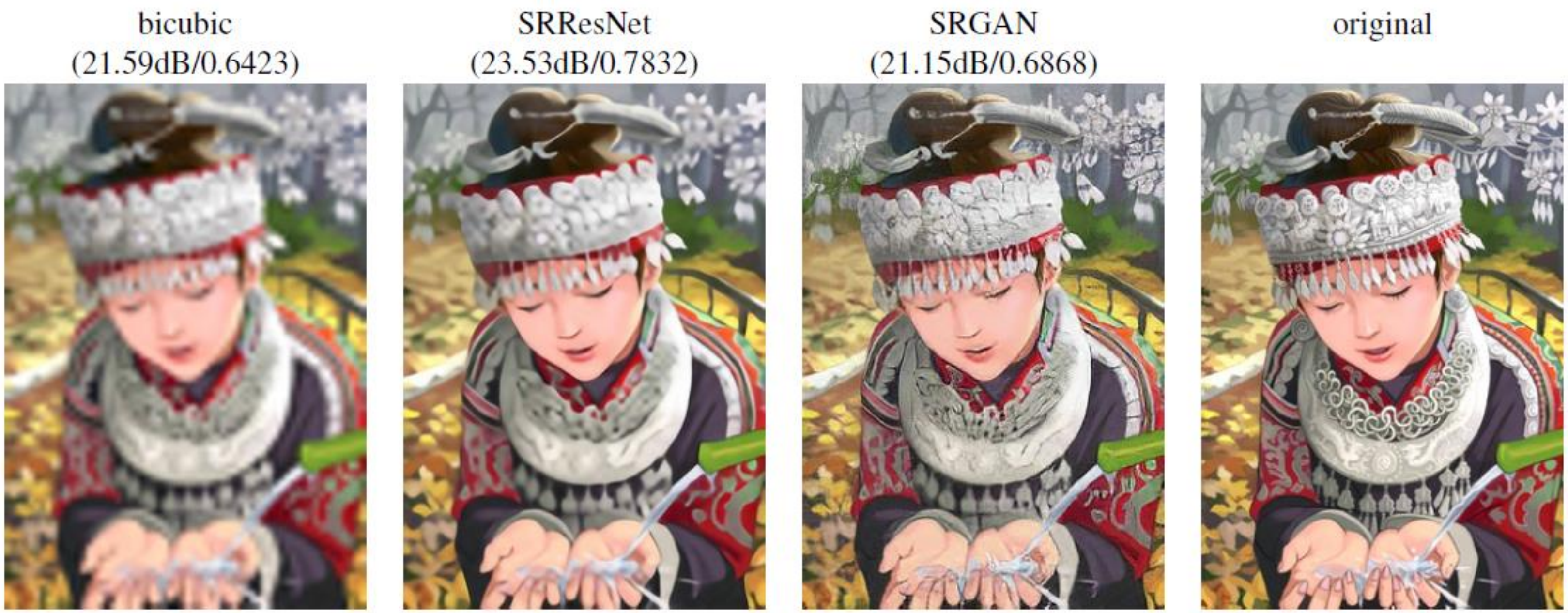
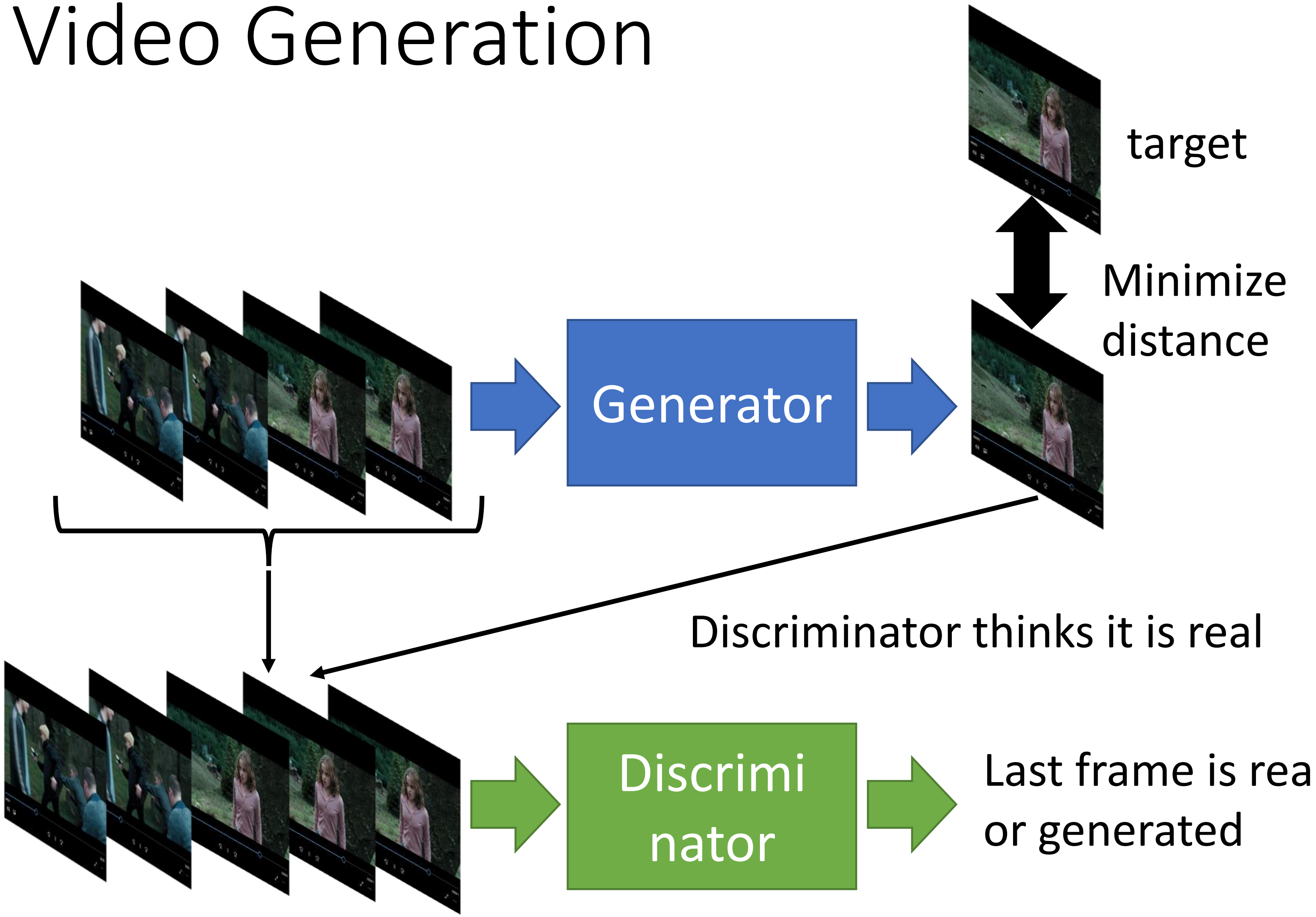
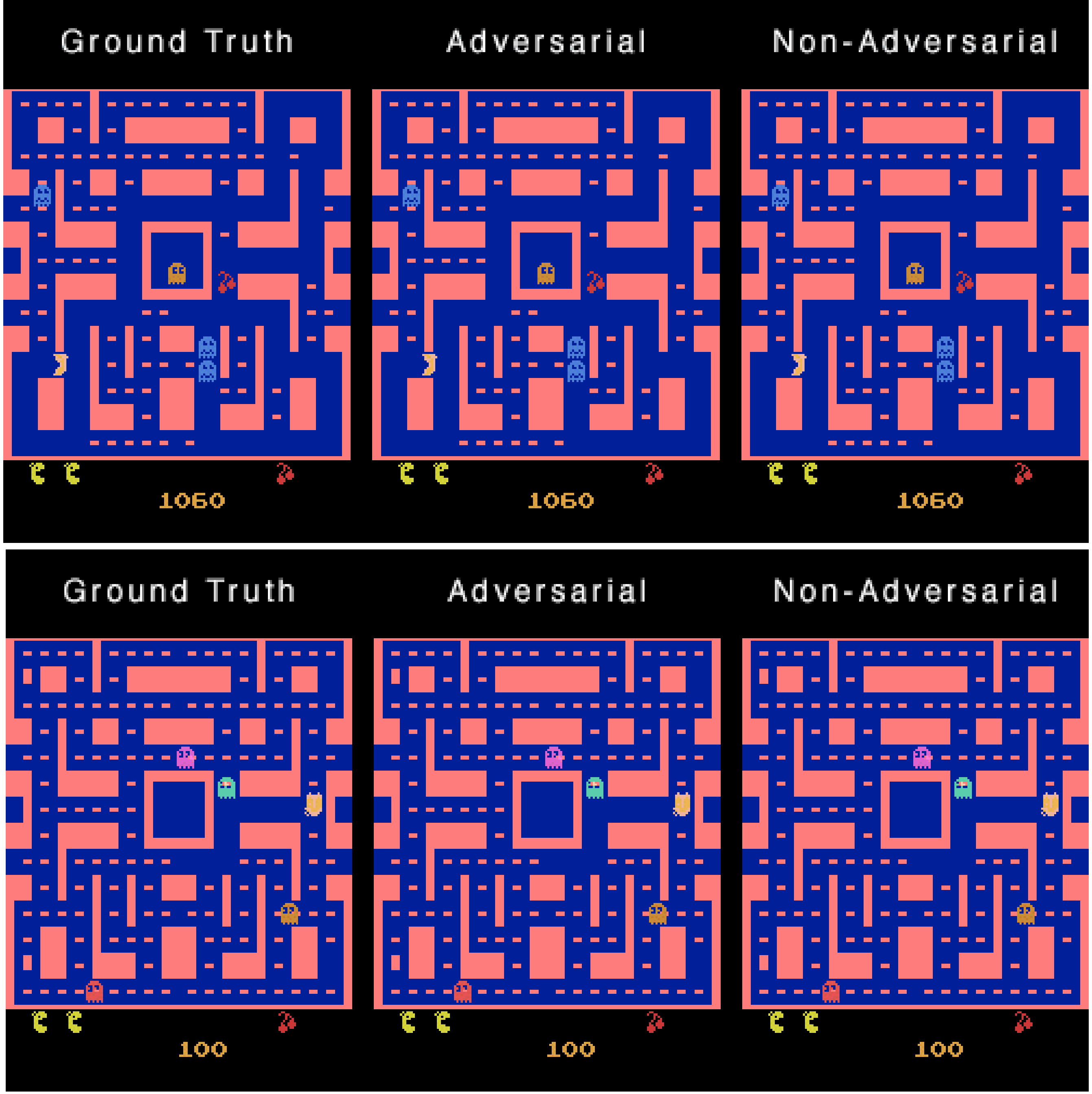


Figure 2: From left to right: bicubic interpolation, deep residual network optimized for MSE, deep residual generative adversarial network optimized for a loss more sensitive to human perception, original HR image. Corresponding PSNR and SSIM are shown in brackets. [4× upscaling]

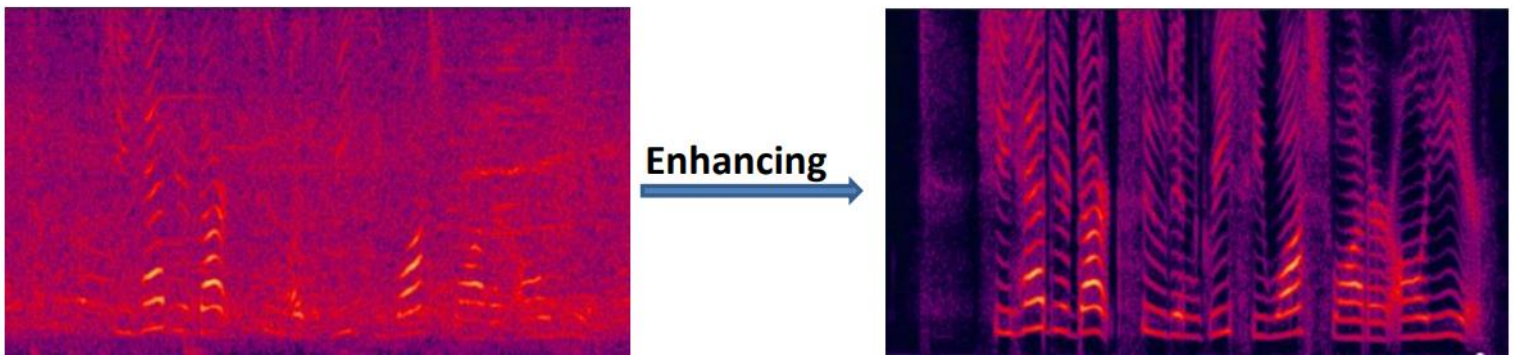
Video Generation



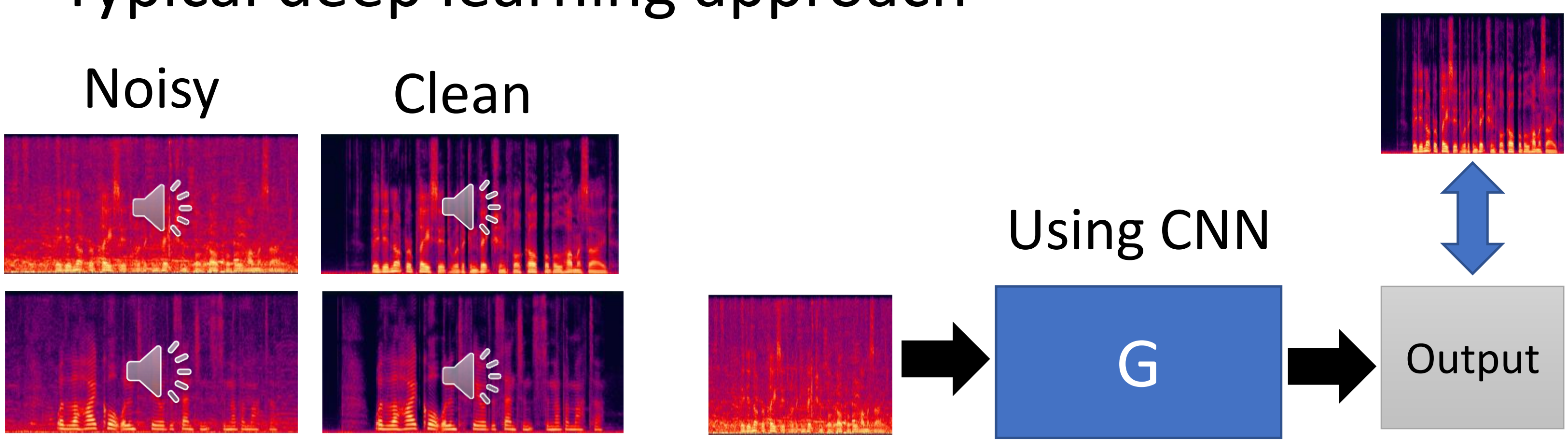


https://github.com/dyelax/Adversarial_Video_Generation

Speech Enhancement

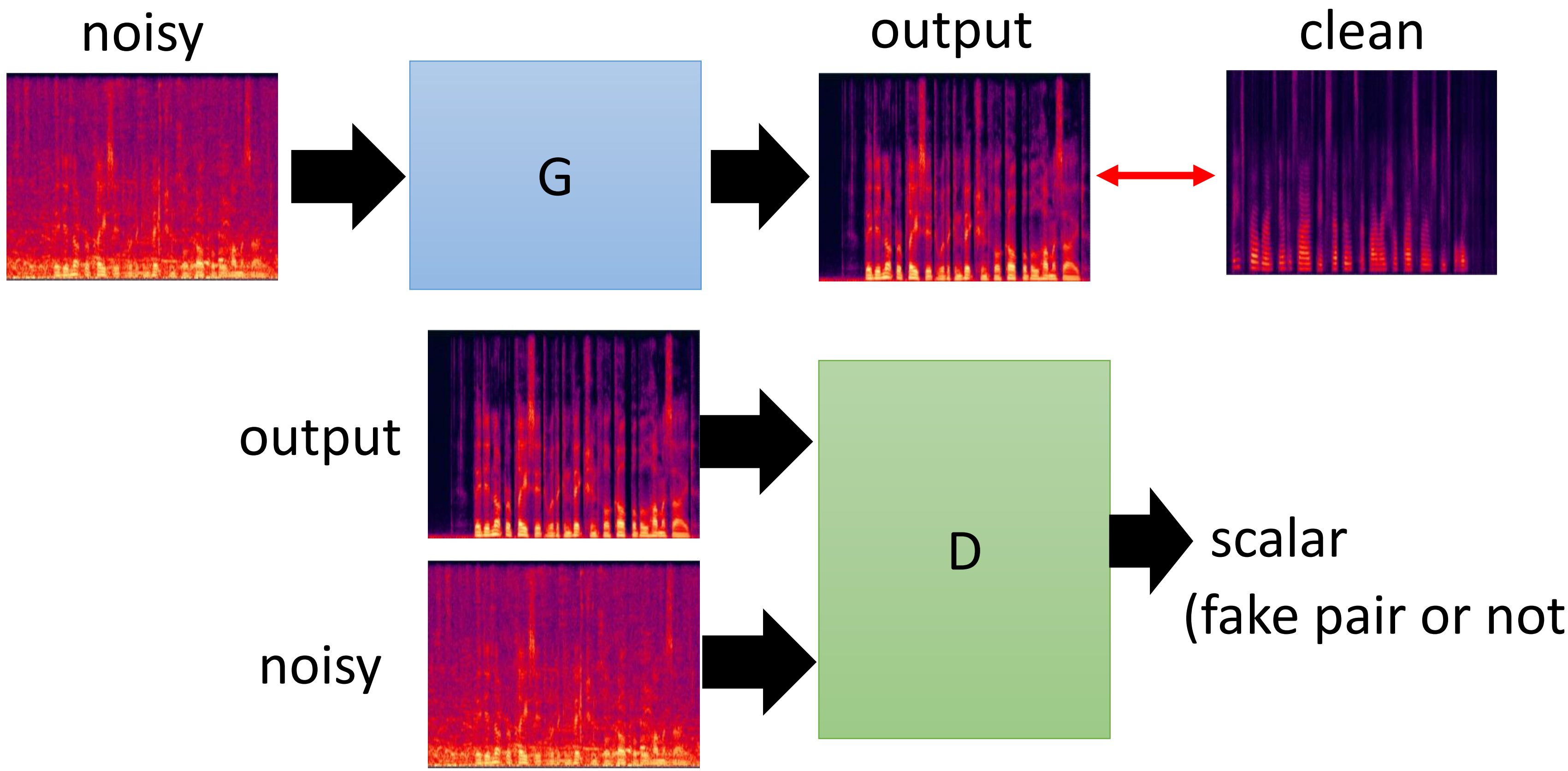


- Typical deep learning approach



Speech Enhancement

- Conditional GAN



Outline

Basic Idea of GAN

When do we need GAN?

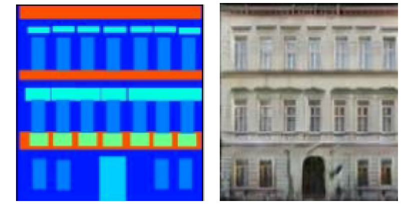
GAN as structured learning algorithm

Conditional Generation by GAN

- Modifying input code
- Paired data
- $\boxed{\text{Unpaired data}}$
- Application: Intelligent Photoshop

Cycle GAN, Disco GAN

paired data



Transform an object from one domain to another ***without paired data***

photo van Gogh

Domain X

Domain Y

Cycle GAN

https://arxiv.org/abs/1703.10593
https://junyanz.github.io/CycleGAN/

Domain X

Domain Y

Domain X

ignore input

$G_{X \to Y}$

Become similar to domain Y

Not what we want

D_Y

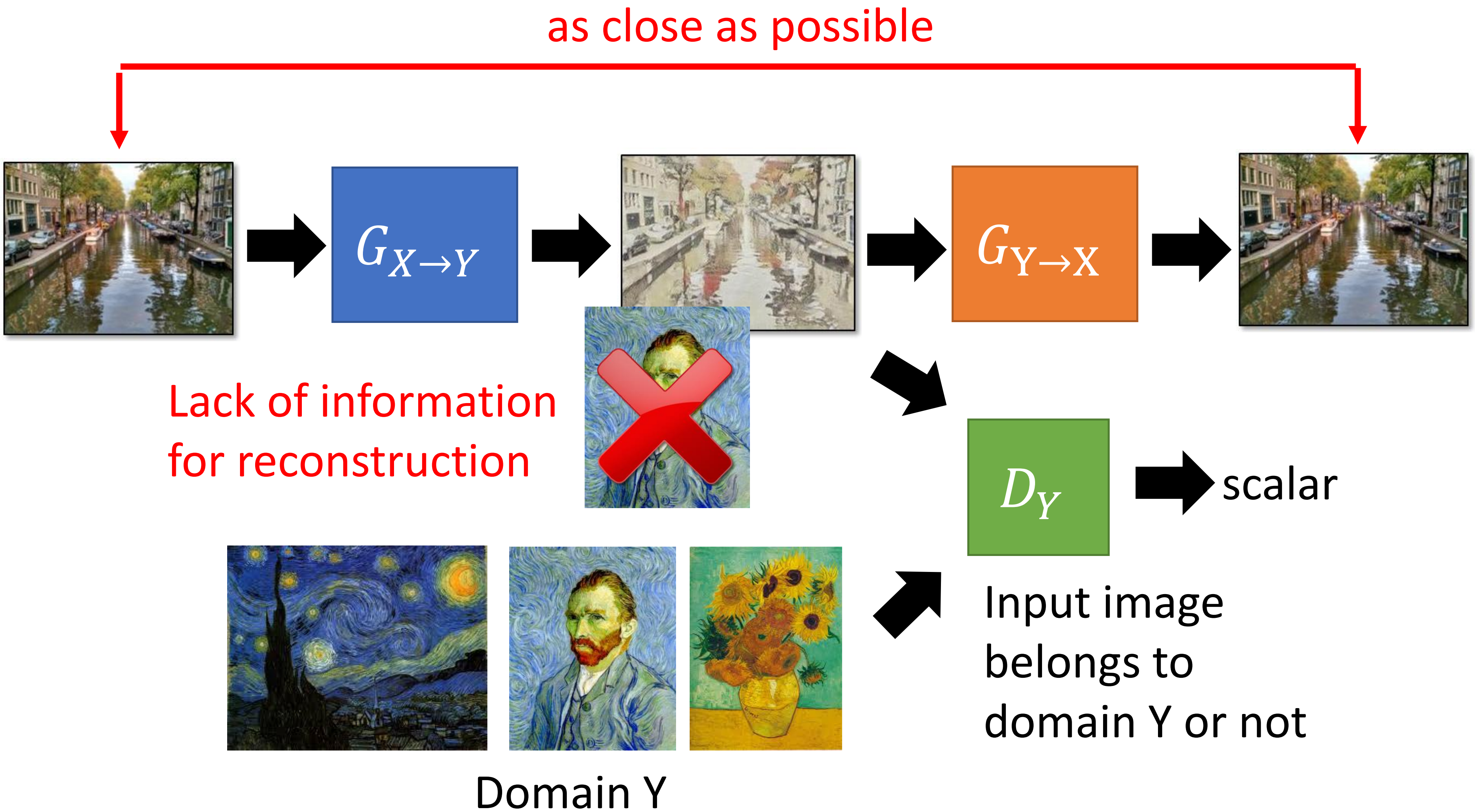
scalar

Input image belongs to domain Y or not

Domain Y

Cycle GAN

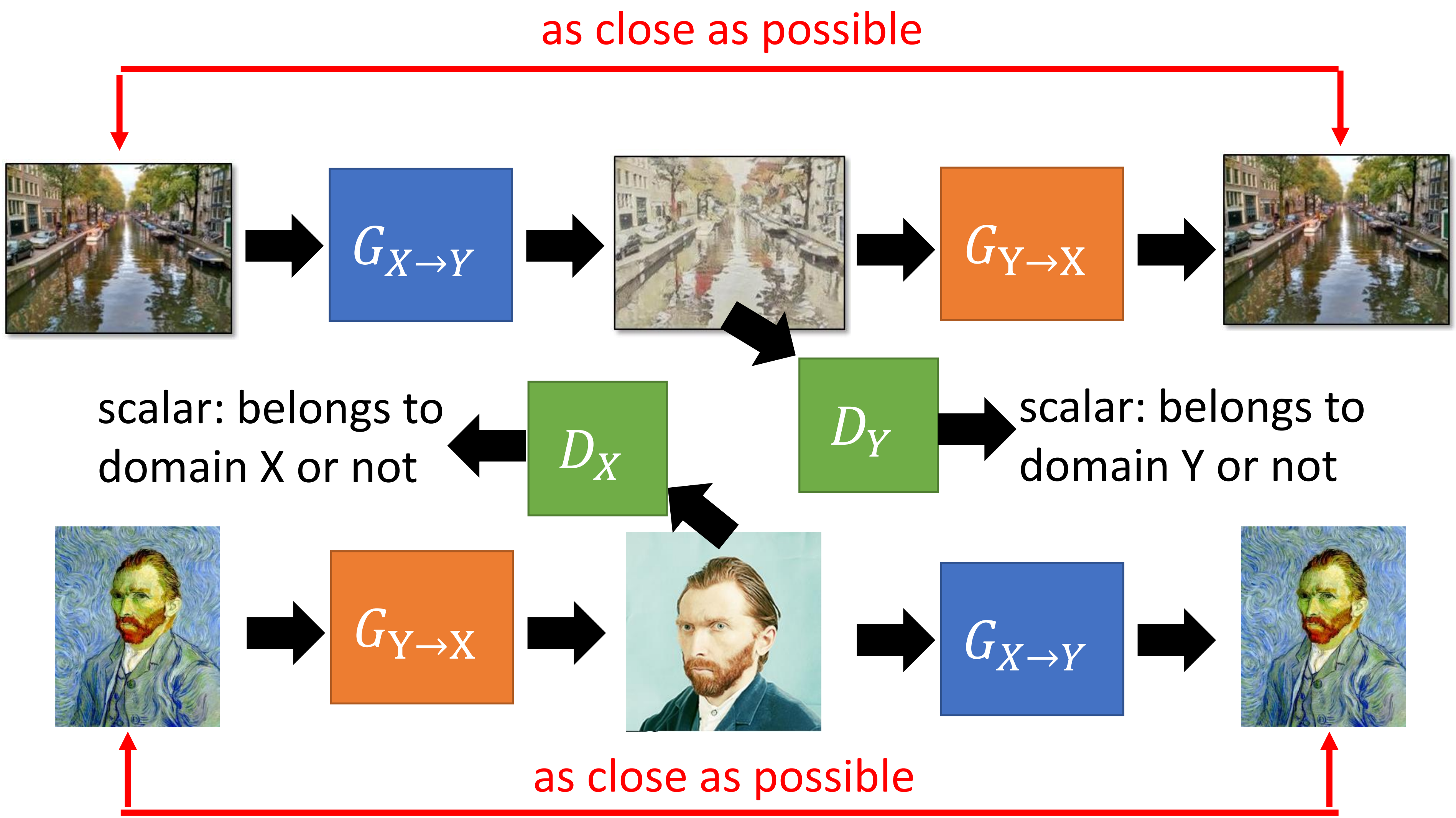
Domain X
Domain Y



as close as possible
$G_{X \to Y}$
$G_{Y \to X}$
Lack of information
for reconstruction
D_Y
scalar
Input image
belongs to
domain Y or not
Domain Y

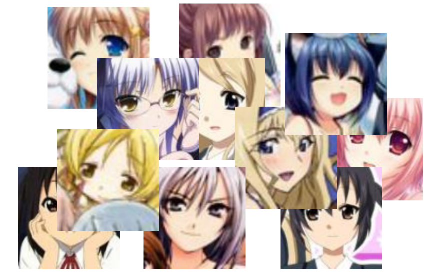
c.f. Dual Learning

Cycle GAN



動畫化的世界

input



output ***domain***

- Using the code: https://github.com/Hi-king/kawaii_creator
- It is not cycle GAN, Disco GAN

Outline

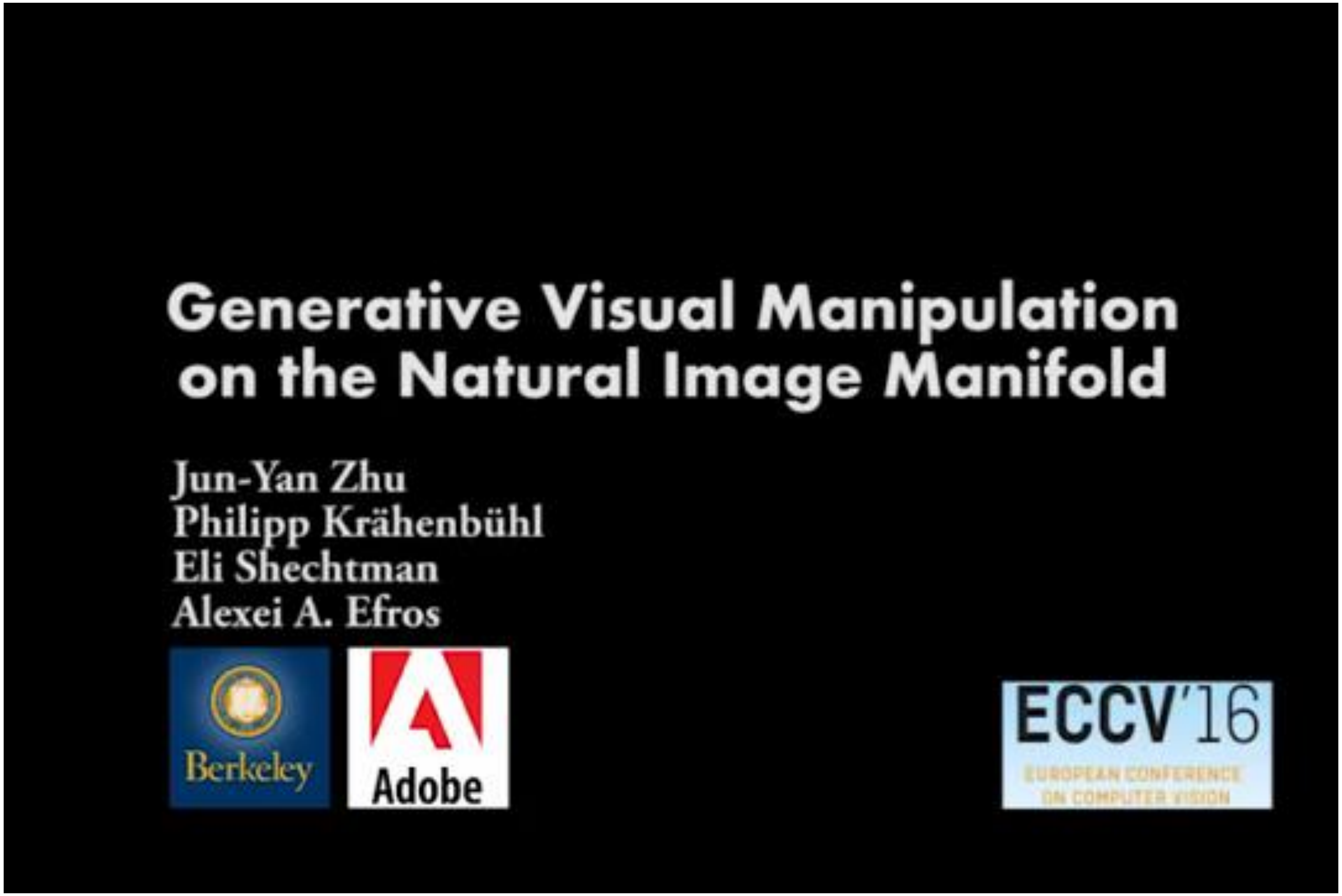
Basic Idea of GAN

When do we need GAN?

GAN as structured learning algorithm

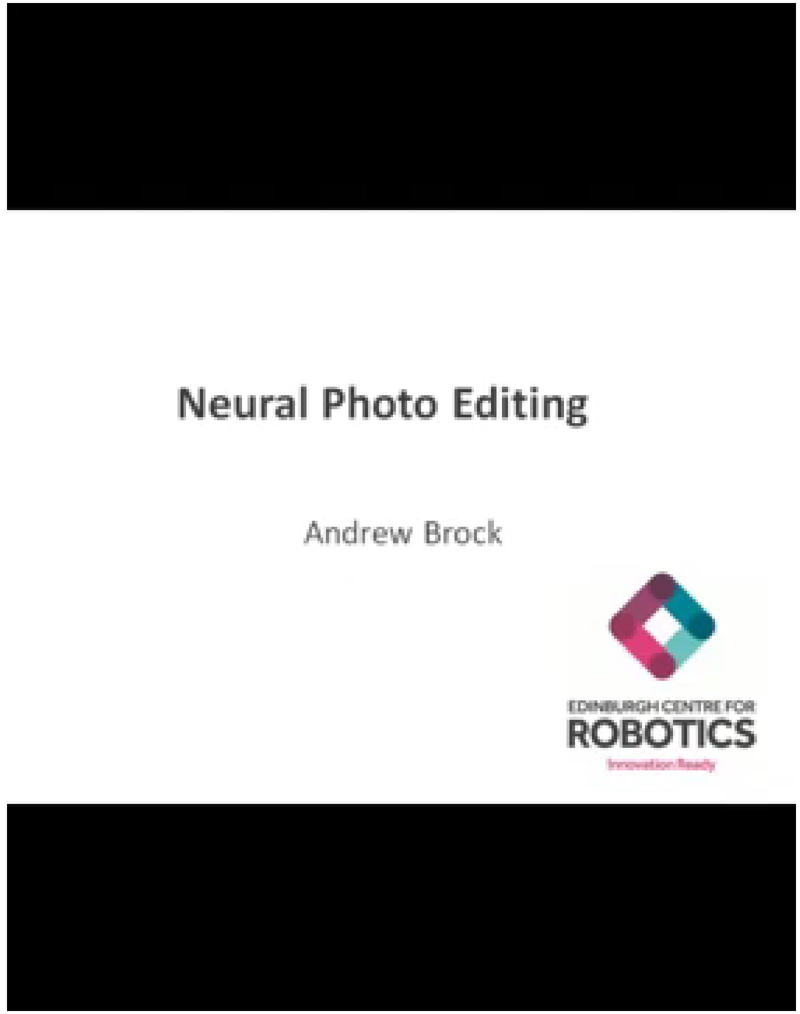
Conditional Generation by GAN

- Modifying input code
- Paired data
- Unpaired data
- $\boxed{\text{Application: Intelligent Photoshop}}$



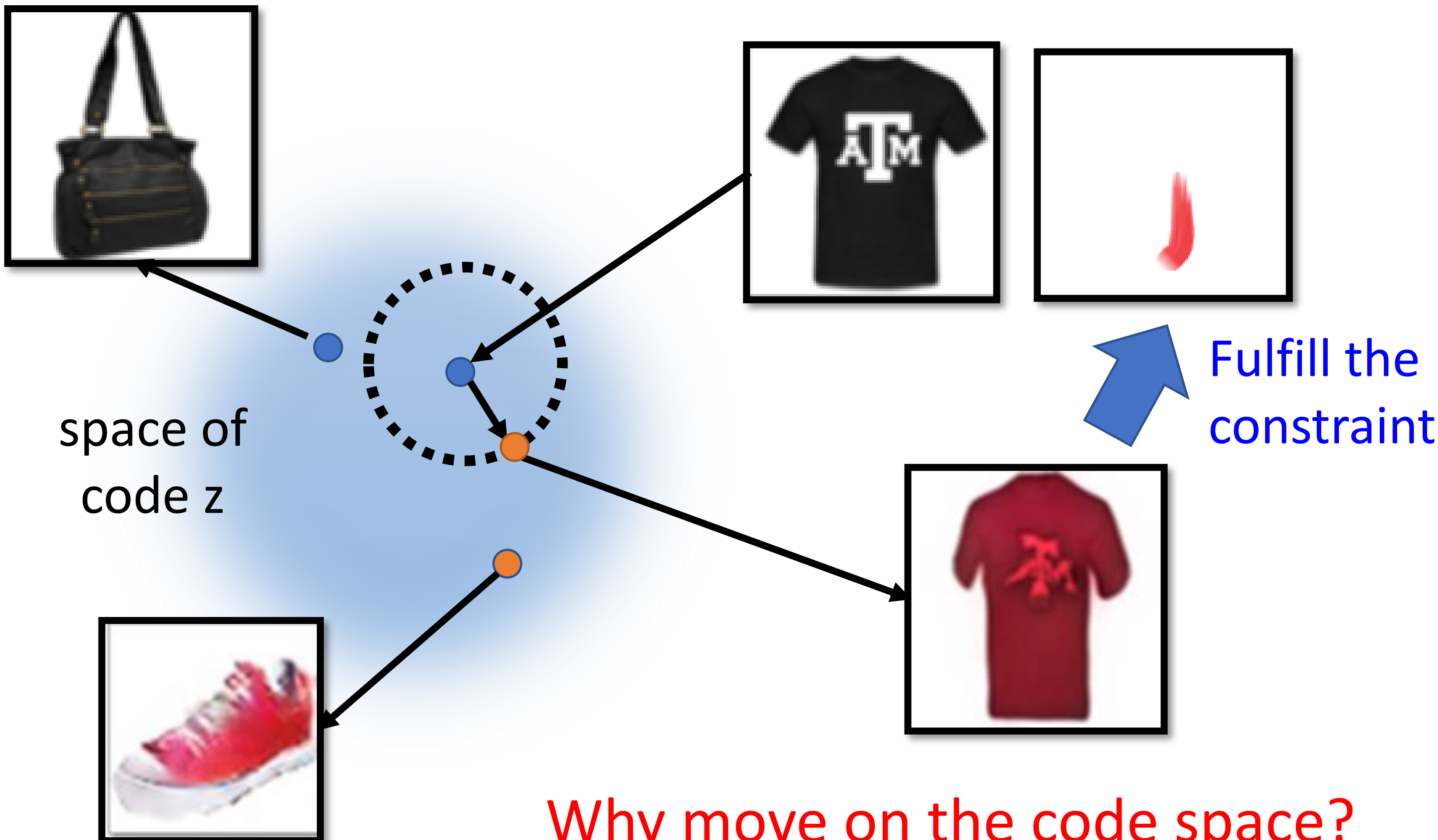
https://www.youtube.com/watch?v=9c4z6YsBGQ0

Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman and Alexei A. Efros. "Generative Visual Manipulation on the Natural Image Manifold", ECCV, 2016.



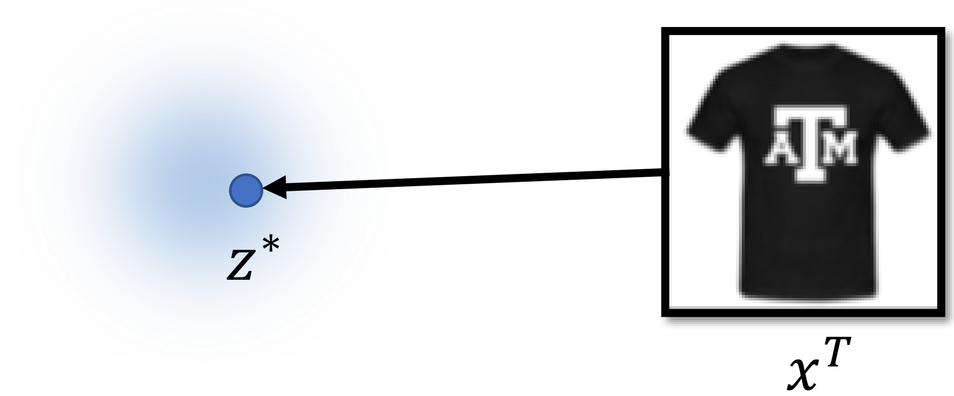
Andrew Brock, Theodore Lim, J.M. Ritchie, Nick Weston, **Neural Photo Editing with Introspective Adversarial Networks,** arXiv preprint, 2017

Basic Idea



Why move on the code space?

Back to z



- ***Method 1***

$$z^* = arg\min_z \underline{L(G(z), x^T)}$$

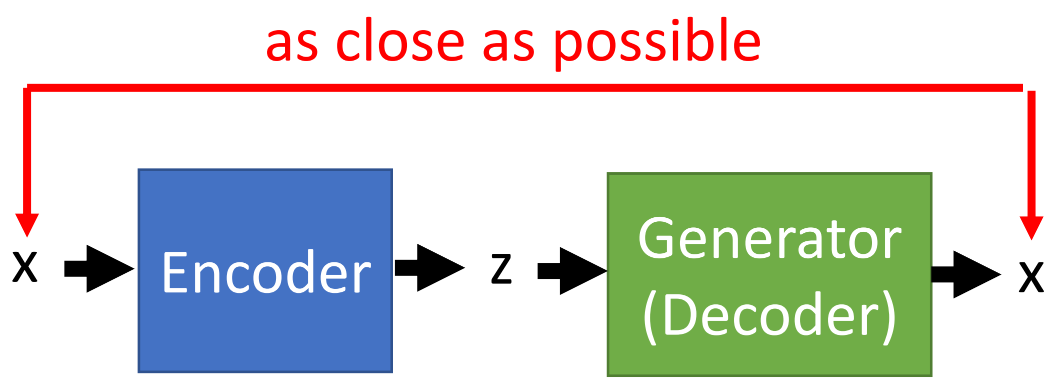
Gradient Descent

Difference between G(z) and x^T

- Pixel-wise
- By another network

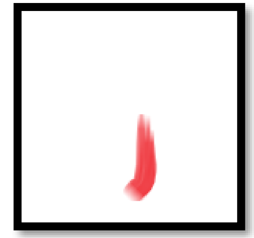
- ***Method 2***

- ***Method 3***



Using the results from ***method 2*** as the initialization of ***method 1***

Editing Photos



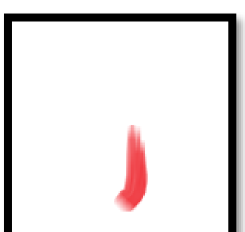
- z_0 is the code of the input image

image

$$z^* = arg\min_z U\left(G(z)\right) + \lambda_1 \|z - z_0\|^2 - \lambda_2 D\left(G(z)\right)$$

Using discriminator to check the image is realistic or not

Not too far away from the original image



Does it fulfill the constraint of editing?

Concluding Remarks

Basic Idea of GAN

When do we need GAN?

GAN as structured learning algorithm

Conditional Generation by GAN

- Modifying input code
- Paired data
- Unpaired data
- Application: Intelligent Photoshop