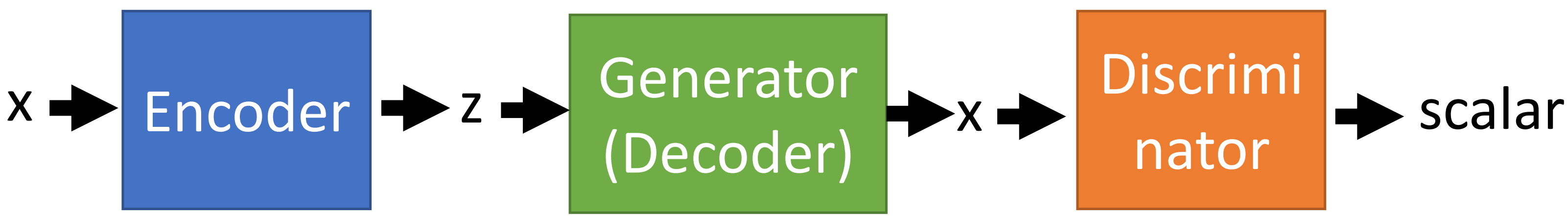
GAN and Feature Representation

Hung-yi Lee

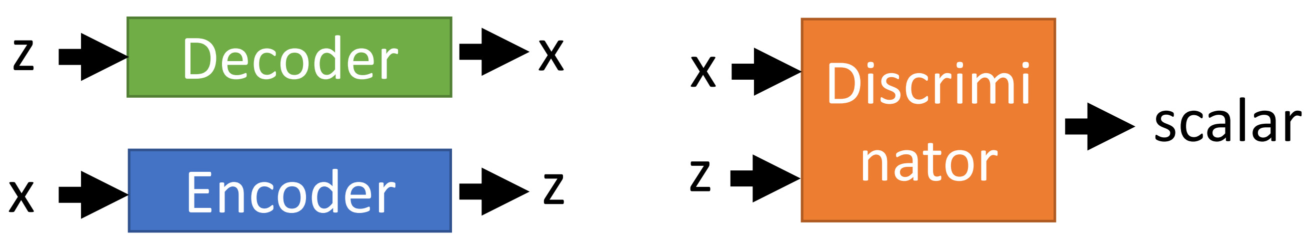
Outline

- GAN+Autoencoder



- InfoGAN

- BiGAN

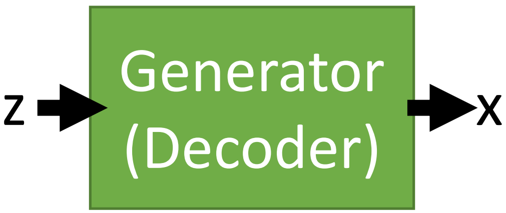


GAN + Autoencoder

Photo Editing

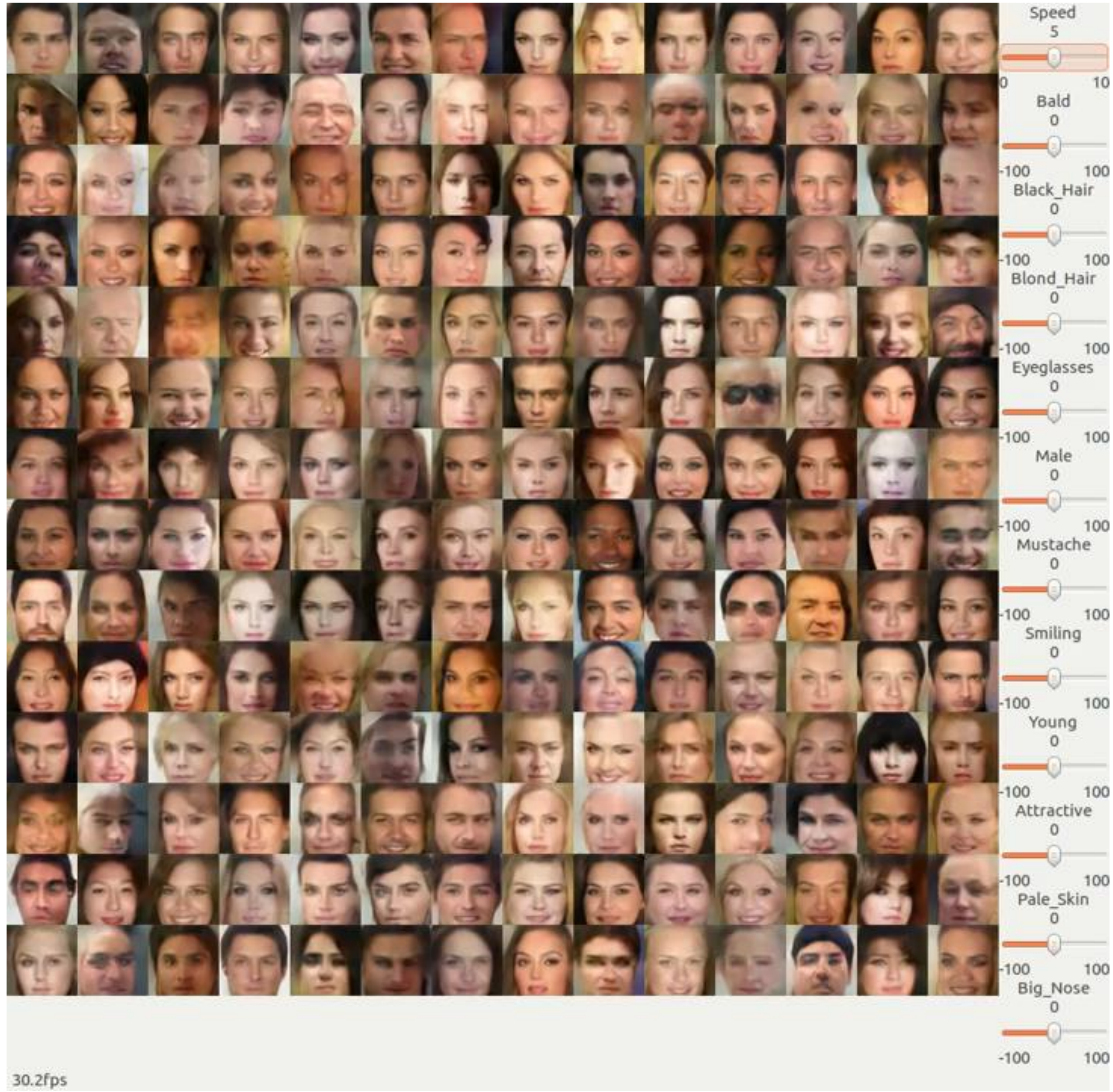
https://devblogs.nvidia.com/parallelforall/photo-editing-generative-adversarial-networks-2/

Photo Editing



We can tune z to edit image x

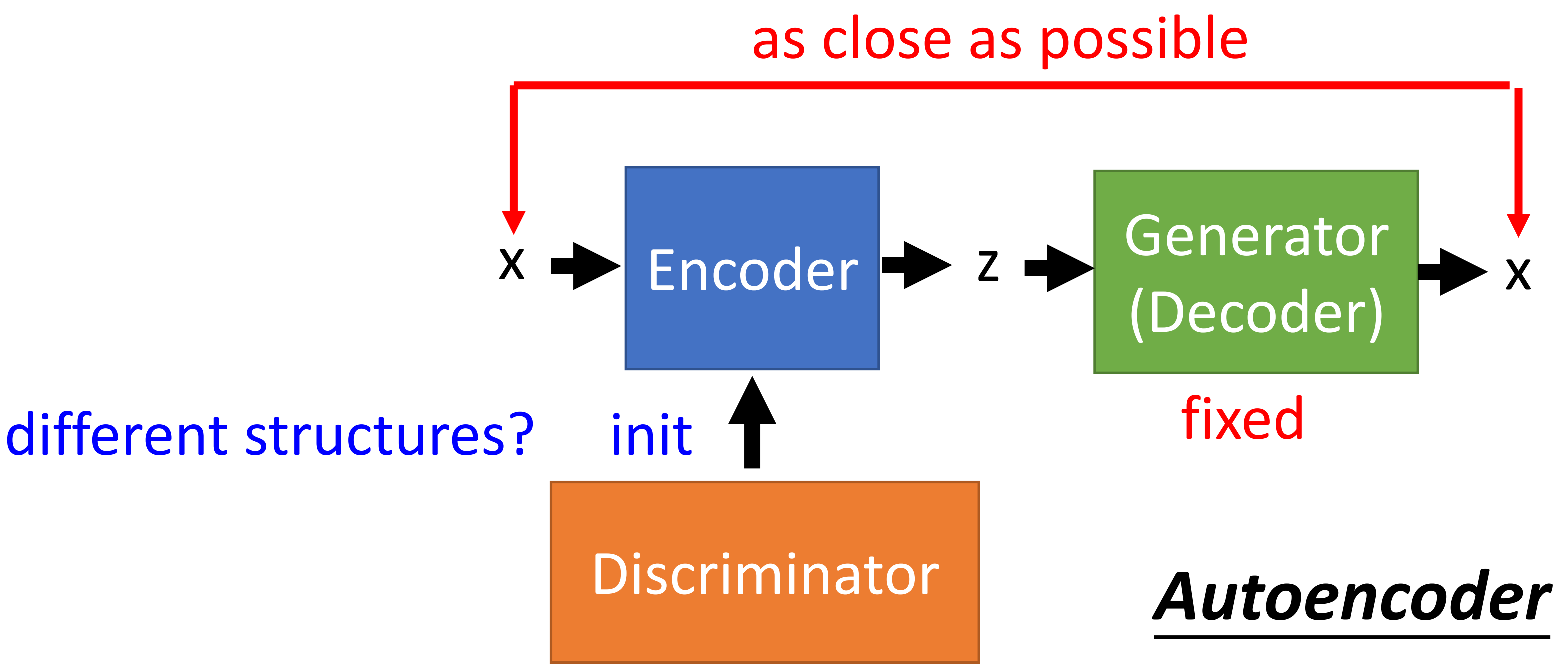
How to modify a specific attribute?

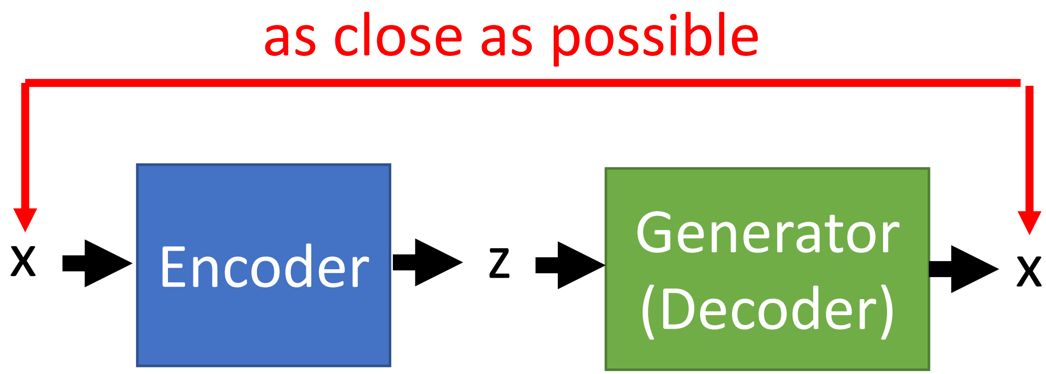


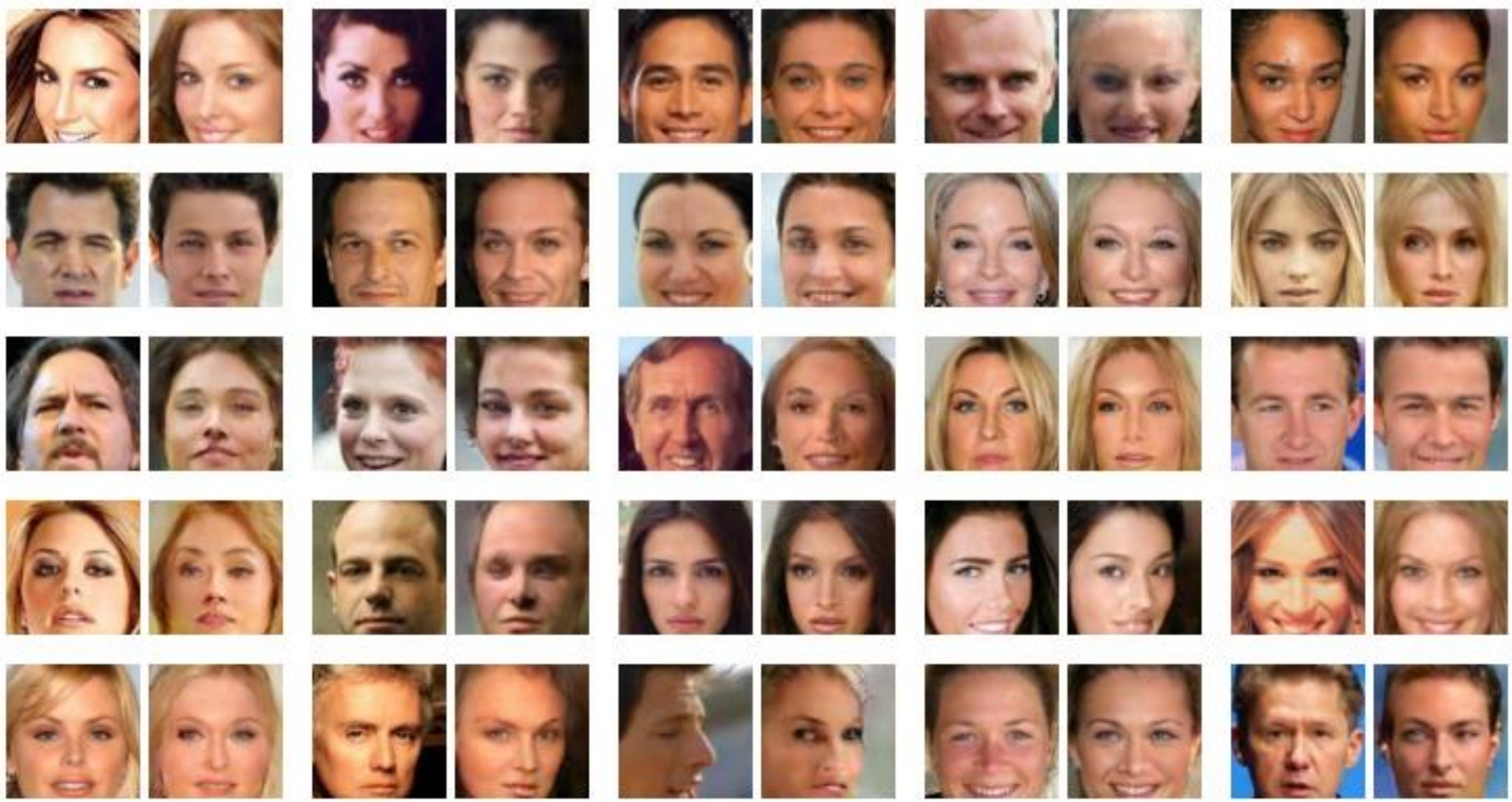
https://www.youtube.com/watch?v=kPEIJJsQr7U

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)

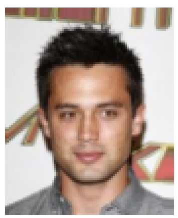



as close as possible
x
Encoder
z
Generator
(Decoder)
x



Attribute Representation

CelebA

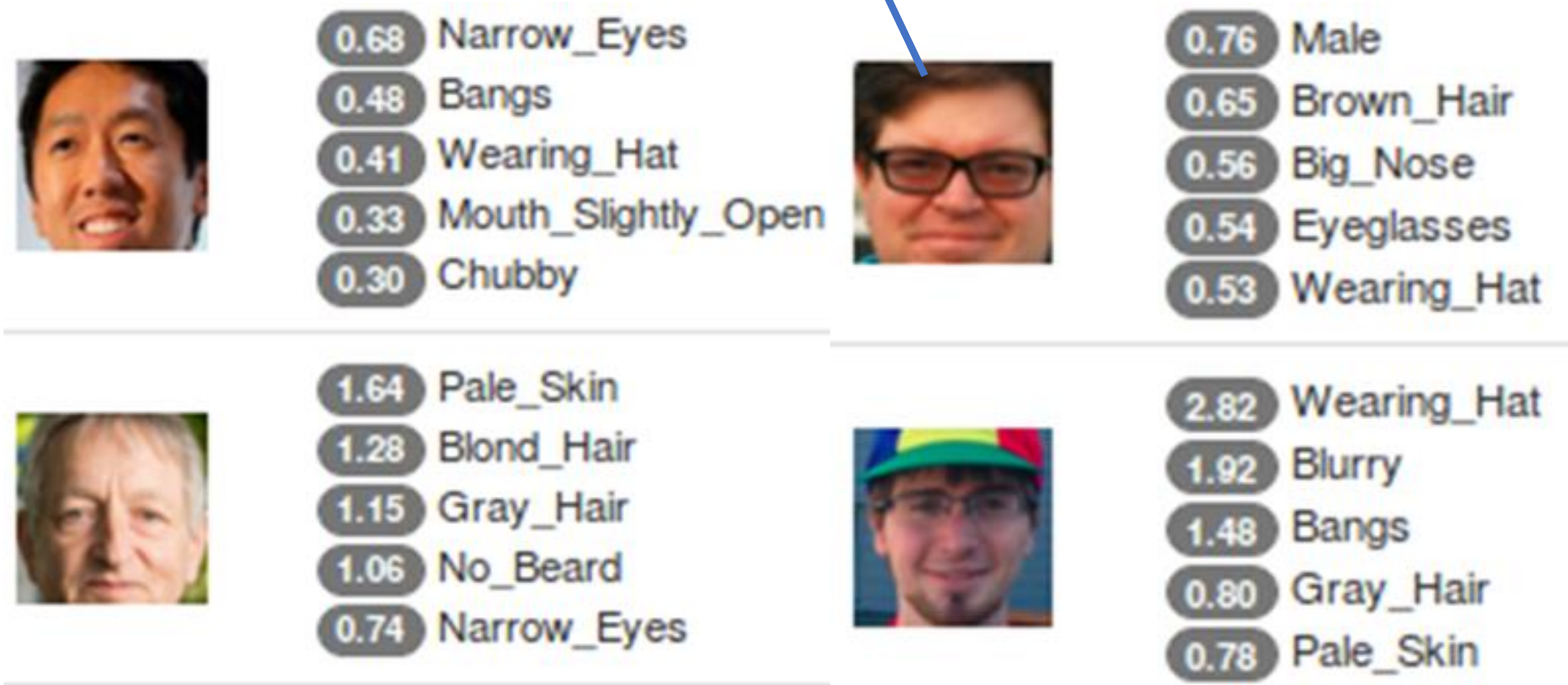
Image	Attributes
	Arched eyebrows, attractive, brown hair, heavy makeup, high cheekbones, mouth slightly open, no beard, pointy nose, smiling, straight hair, wearing earrings, wearing lipstick, young.
	5 o'clock shadows, attractive, bags under eyes, big lips, big nose, black hair, bushy eyebrows, male,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

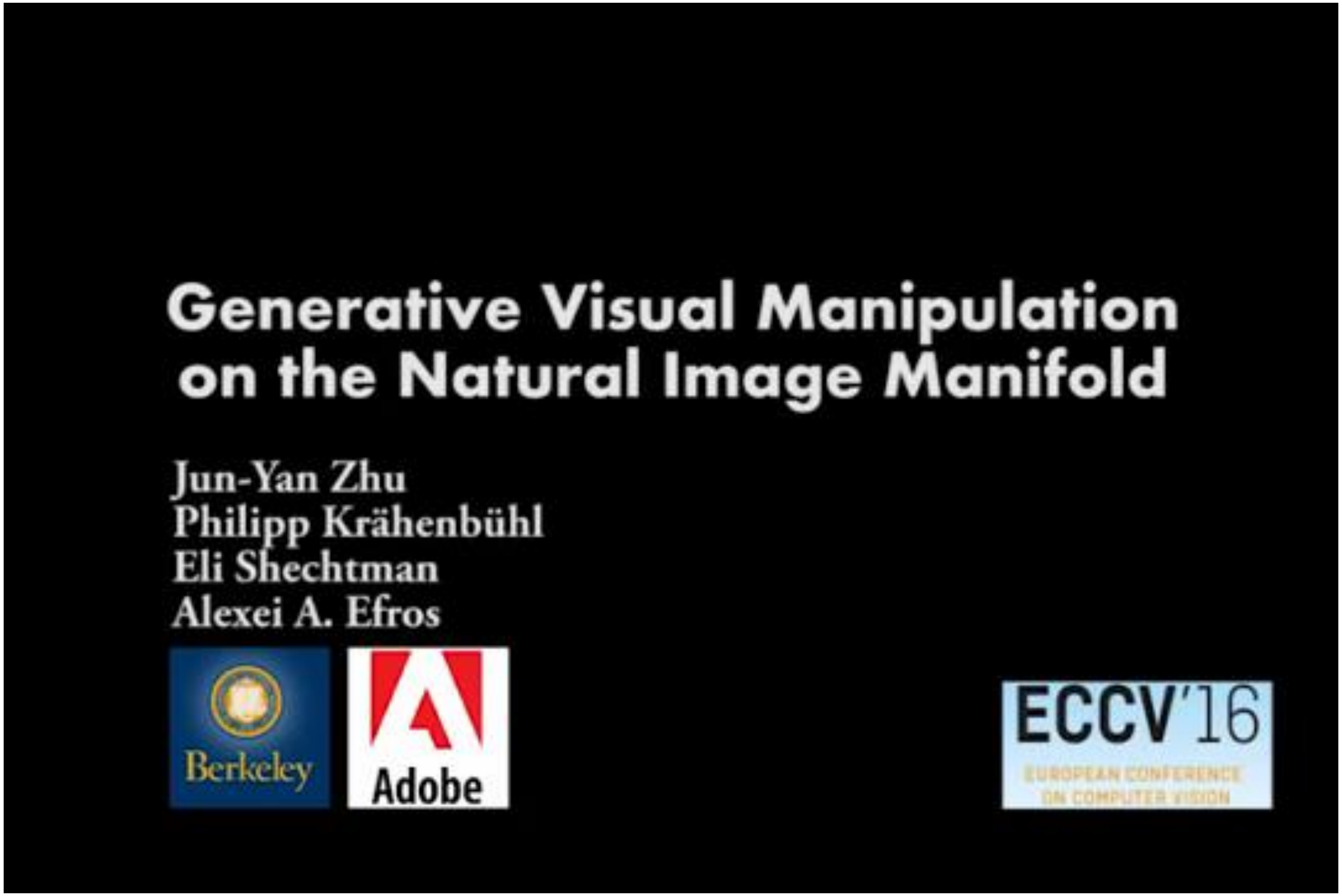
$$z_{male} = \frac{1}{N_1} \sum_{x \in male} En(x) - \frac{1}{N_2} \sum_{x' \notin male} En(x')$$

Female image $x \rightarrow En(x) + z_{male} = z' \rightarrow Gen(z')$ male image

Find the Attributes

$$z \cdot \frac{z_{male}}{\|z_{male}\|} = 0.76$$

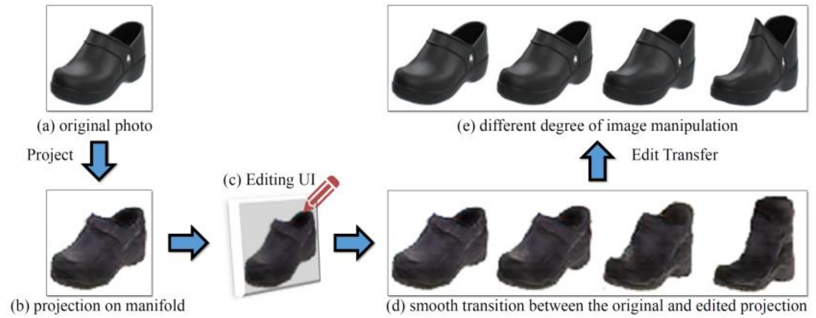


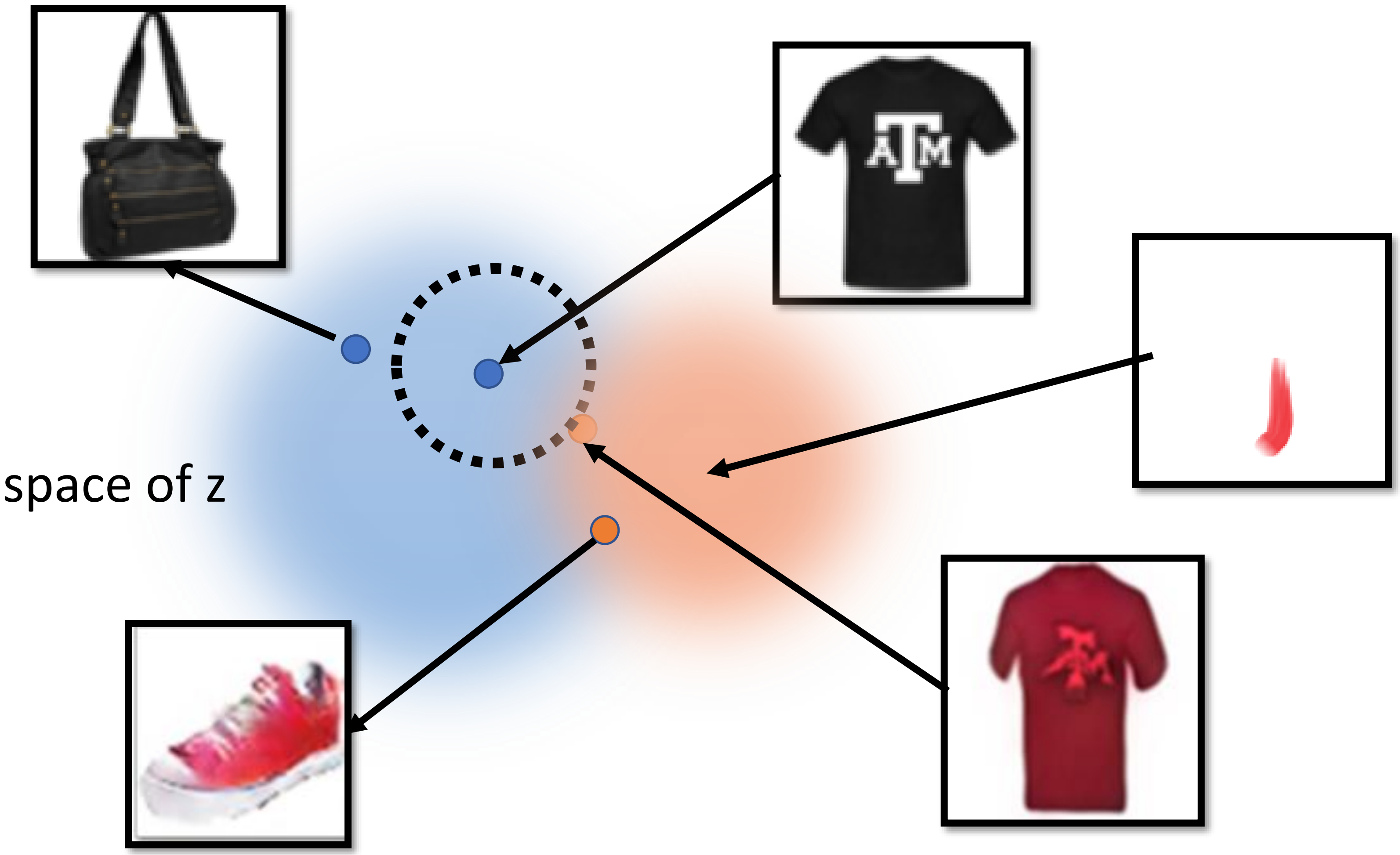


https://www.youtube.com/watch?v=9c4z6YsBGQ0

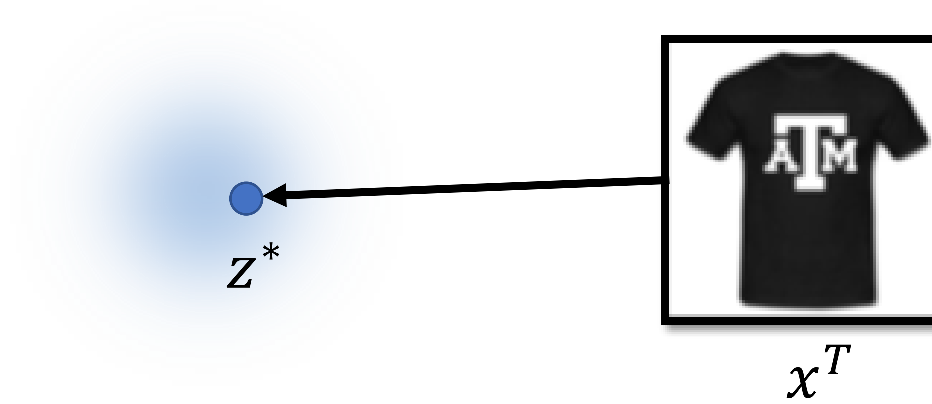
Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman and Alexei A. Efros. "Generative Visual Manipulation on the Natural Image Manifold", ECCV, 2016.

Basic Idea


(a) original photo
Project
(e) different degree of image manipulation
Edit Transfer
(c) Editing UI
(b) projection on manifold
(d) smooth transition between the original and edited projection


space of z

Back to z



- ***Method 1***

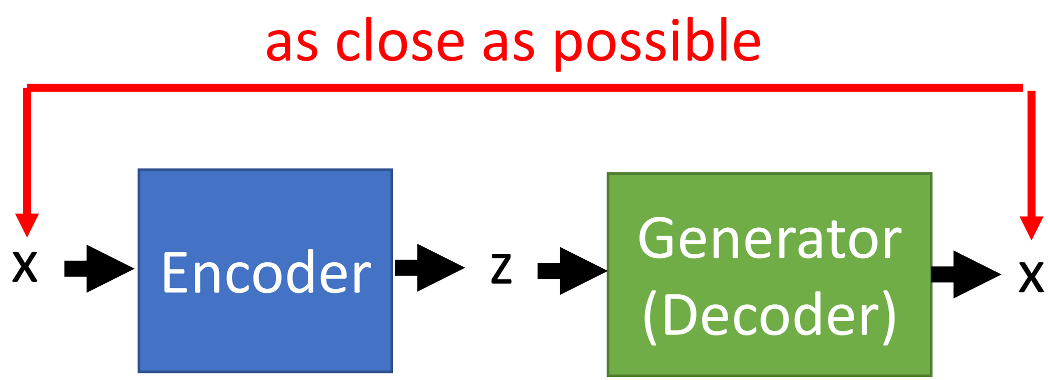
$$z^* = arg \min_z \underline{L(G(z), x^T)}$$

Gradient Descent

Difference between G(z) and x^T

- Pixel-wise
- By another network

- ***Method 2***

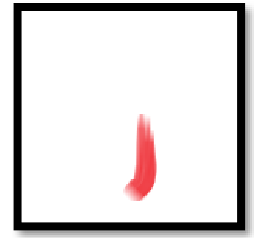


- ***Method 3***

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

Back to z - Results

Editing Photos



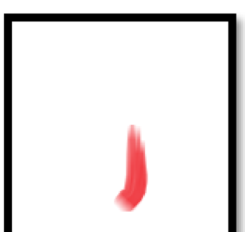
- z_0 is the code of the input image

Using discriminator to check the image is realistic or not

image

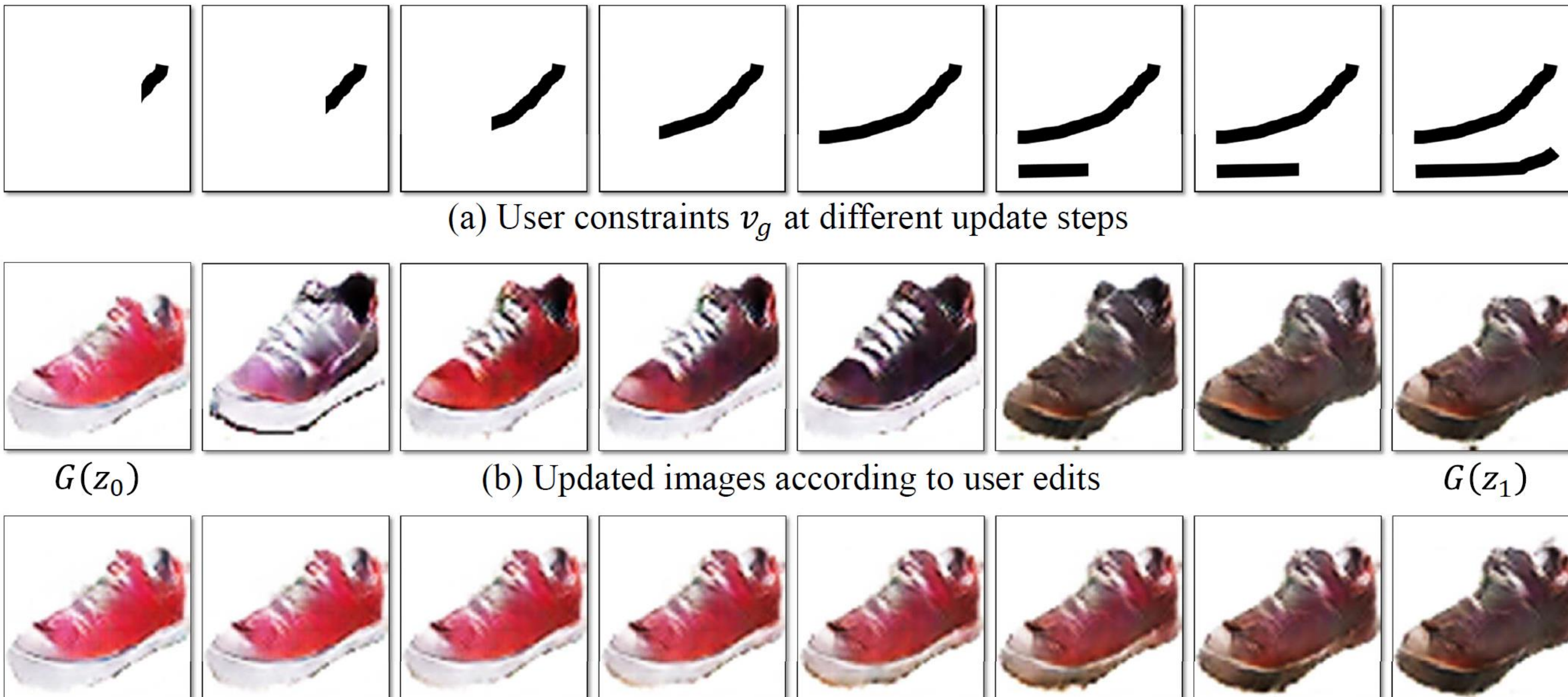
$$z^* = arg\min_{z} U\big(G(z)\big) + \lambda_1 \|z - z_0\|^2 - \lambda_2 D\big(G(z)\big)$$

Not too far away from the original image



Does it fulfill the constraint of editing?

Editing Photos - Results

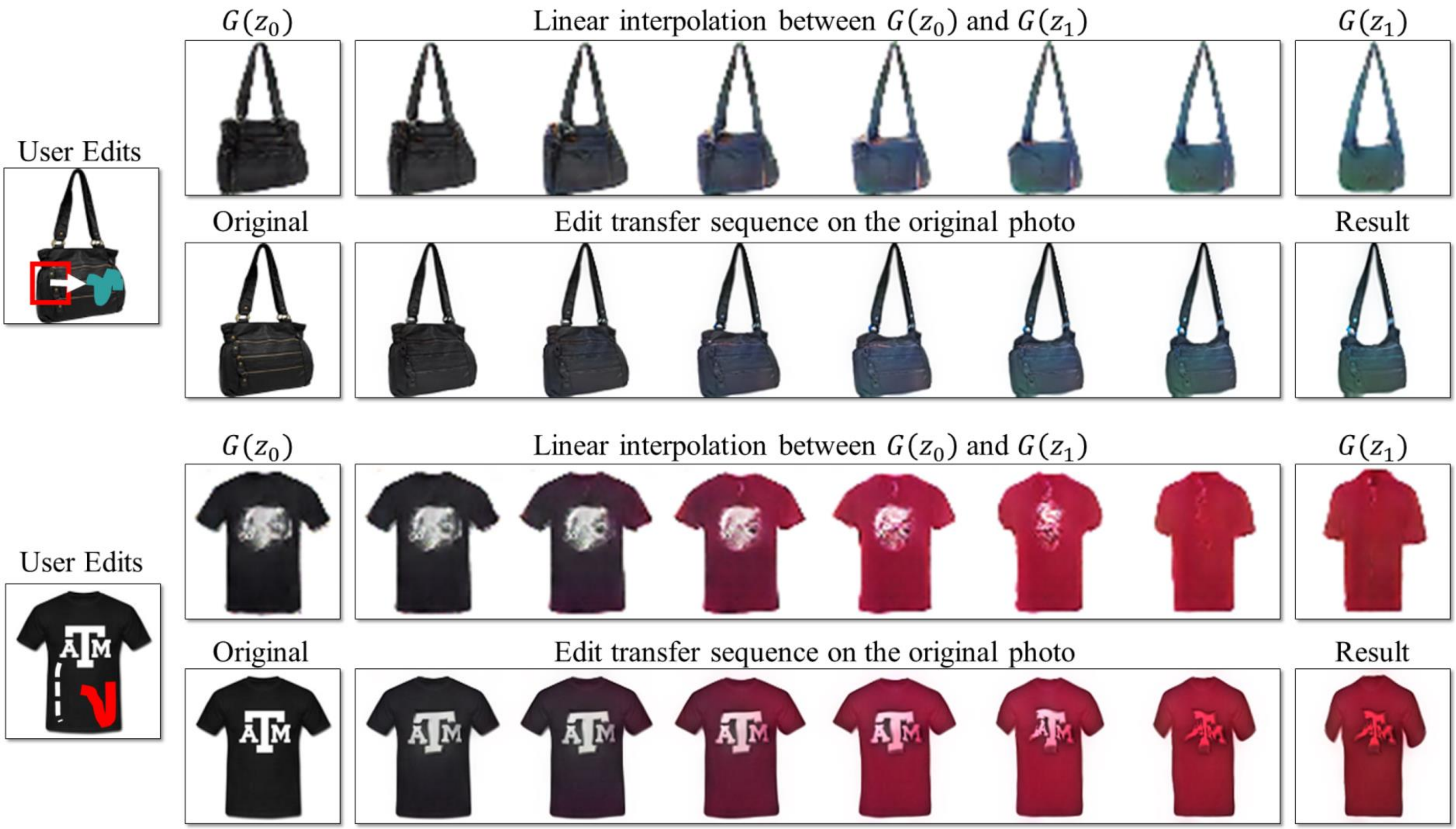


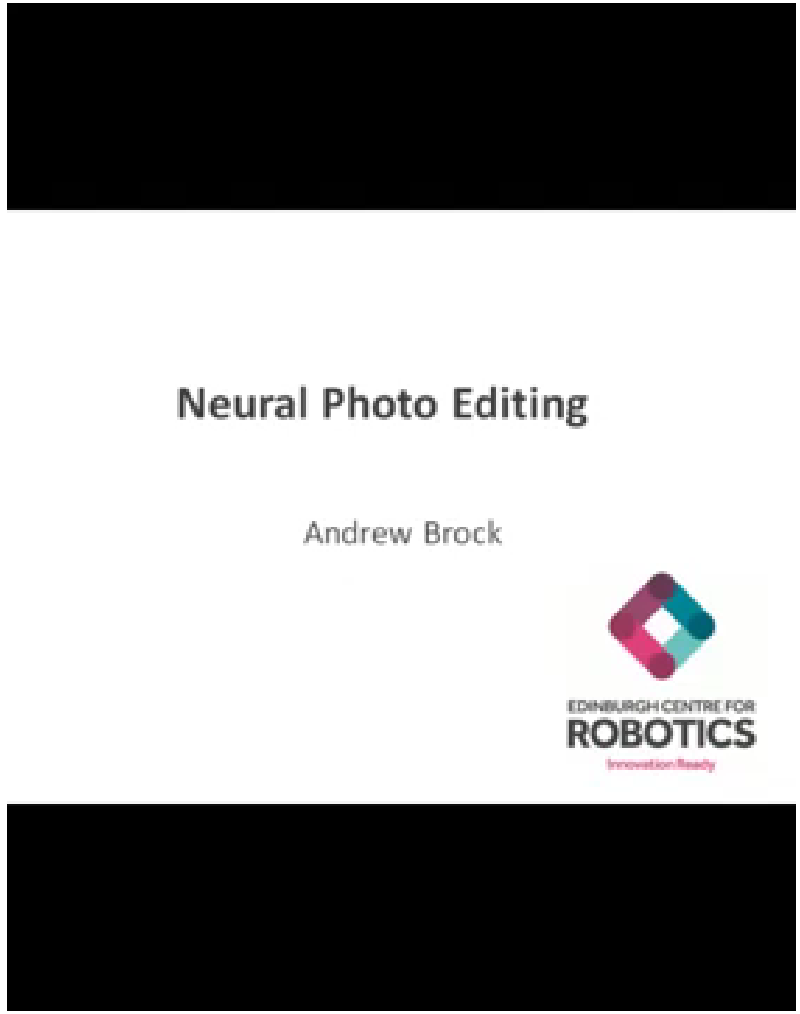
(a) User constraints v_g at different update steps

(b) Updated images according to user edits

(c) Linear interpolation between $G(z_0)$ and $G(z_1)$

Final System





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

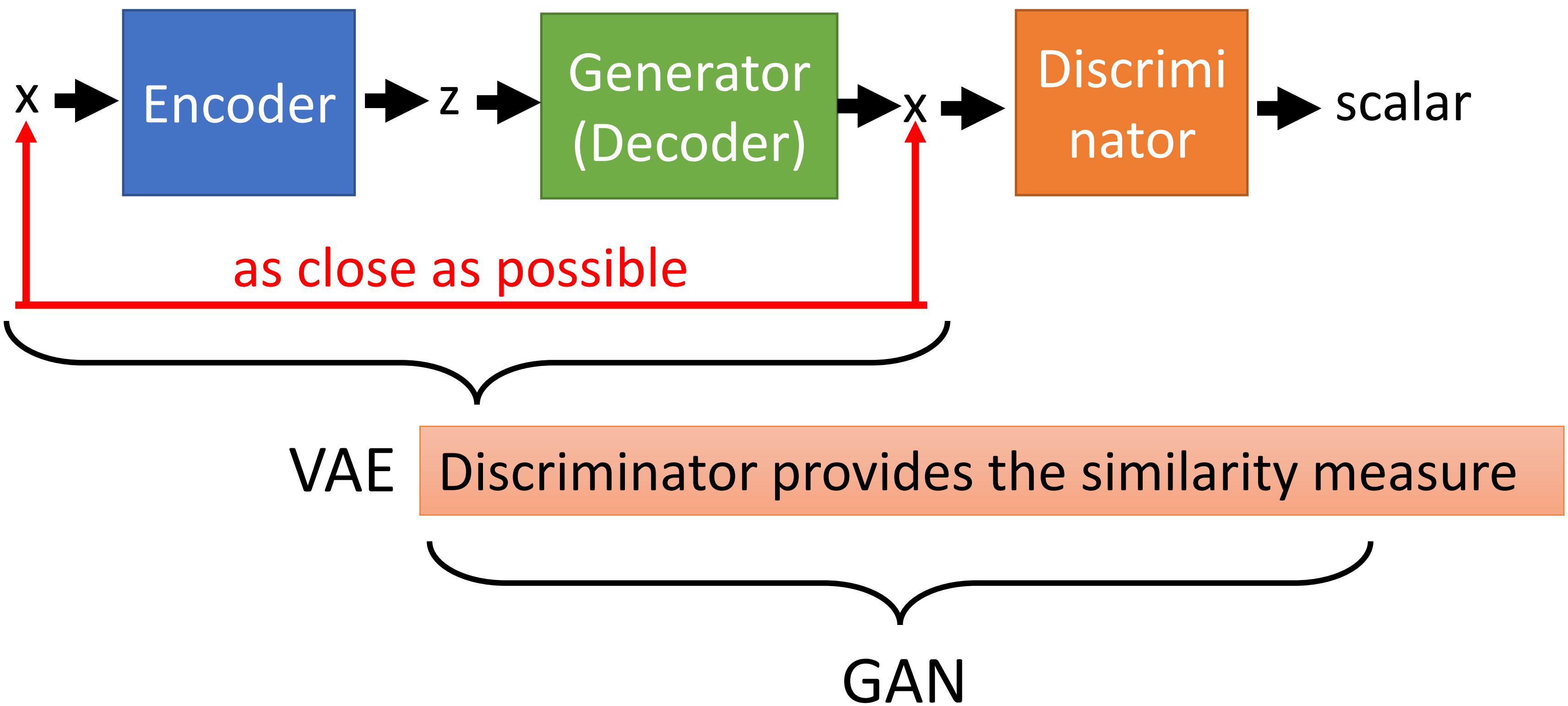
VAE-GAN

Anders Boesen, Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, Ole Winther, “Autoencoding beyond pixels using a learned similarity metric”, ICML. 2016

- Minimize reconstruction error
- z close to normal

- Minimize reconstruction error
- Cheat discriminator

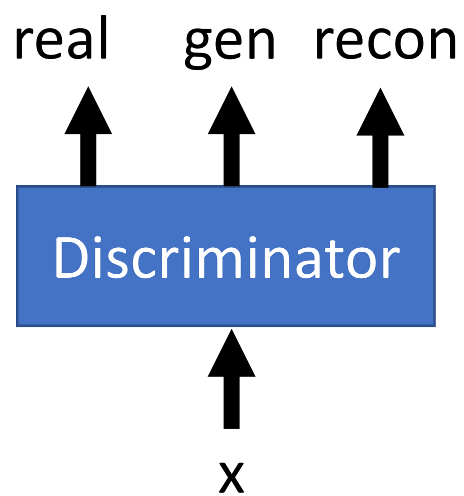
- Discriminate real, generated and reconstructed images



Algorithm

- Initialize En, De, Dis
- In each iteration:
  - Sample M images $x^1, x^2, \cdots, x^M$ from database
  - Generate M codes $\tilde{z}^1, \tilde{z}^2, \cdots, \tilde{z}^M$ from encoder
    - $\tilde{z}^i = En(x^i)$
  - Generate M images $\tilde{x}^1, \tilde{x}^2, \cdots, \tilde{x}^M$ from decoder
    - $\tilde{x}^i = En(\tilde{z}^i)$
  - Sample M codes $z^1, z^2, \cdots, z^M$ from prior P(z)
  - Generate M images $\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^M$ from decoder
    - $\hat{x}^i = En(z^i)$
  - Update En to decrease $\left\|\tilde{x}^i - x^i\right\|$, decrease KL(P($\tilde{z}^i$|$x^i$)||P(z))
  - Update De to decrease $\left\|\tilde{x}^i - x^i\right\|$, increase $Dis\left(\tilde{x}^i\right)$ and $Dis\left(\hat{x}^i\right)$
  - Update Dis to increase $Dis\left(x^i\right)$, decrease $Dis\left(\tilde{x}^i\right)$ and $Dis\left(\hat{x}^i\right)$

Another kind of discriminator:



VAE+GAN - Sample

blurry

sharp

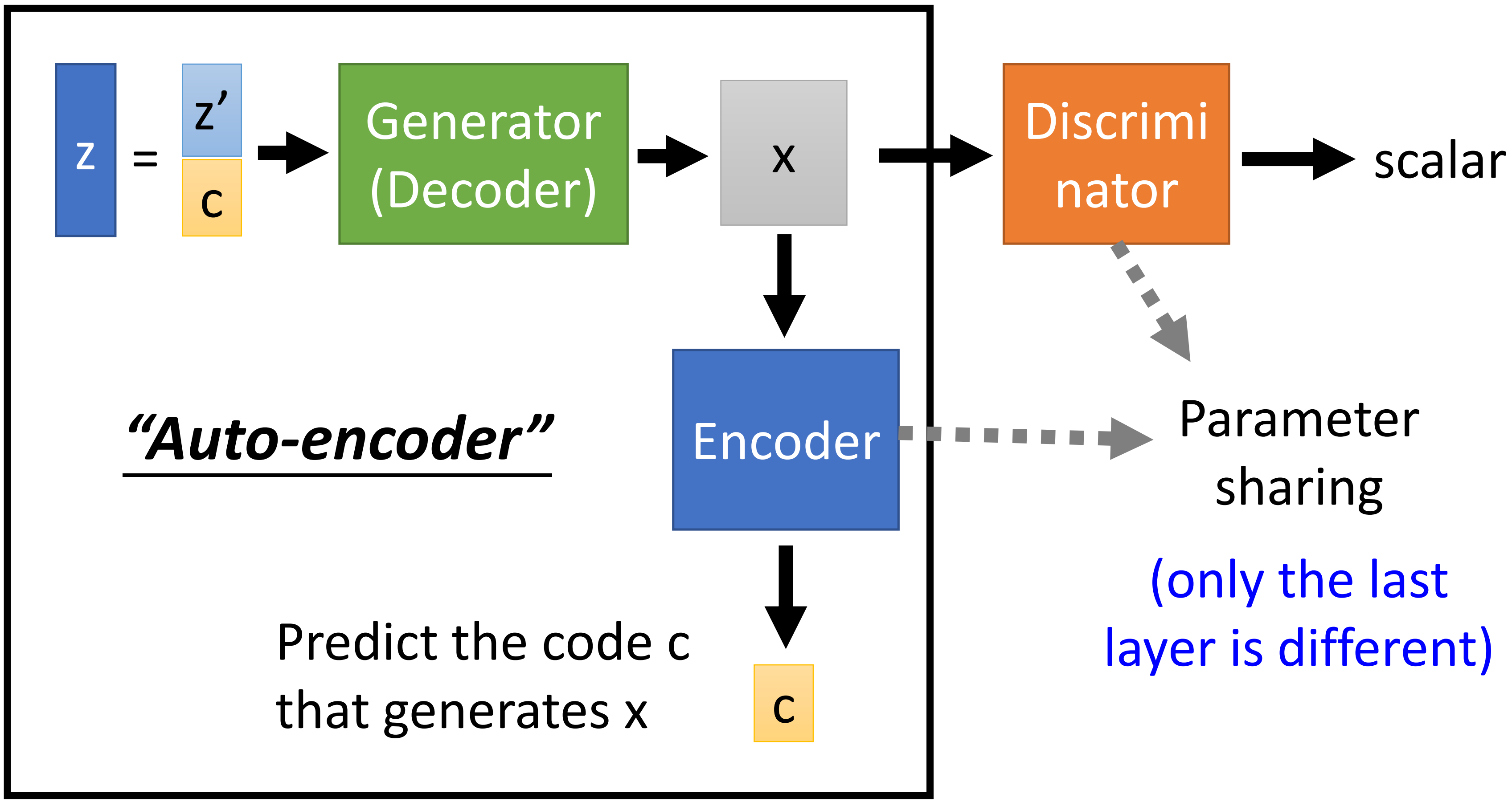
sharp

VAE+GAN - Reconstruction

GAN cannot do reconstruction

InfoGAN

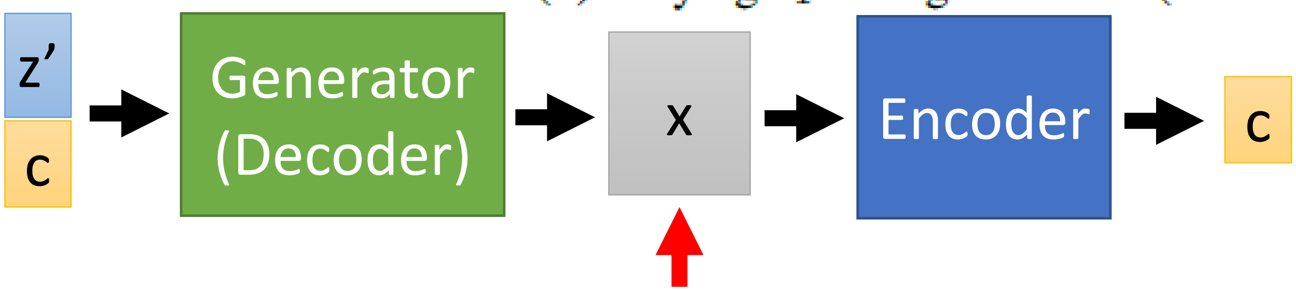
What is InfoGAN?



Motivation

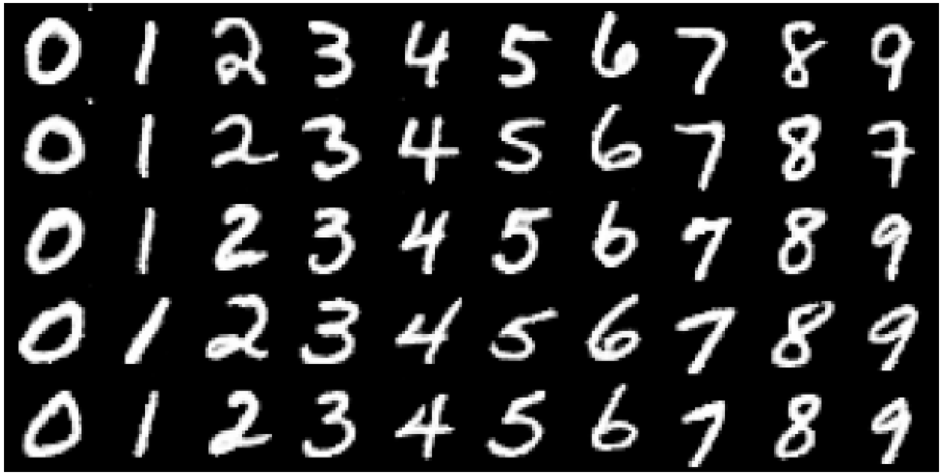
Regular GAN

(b) Varying c_1 on regular GAN (No clear meaning)

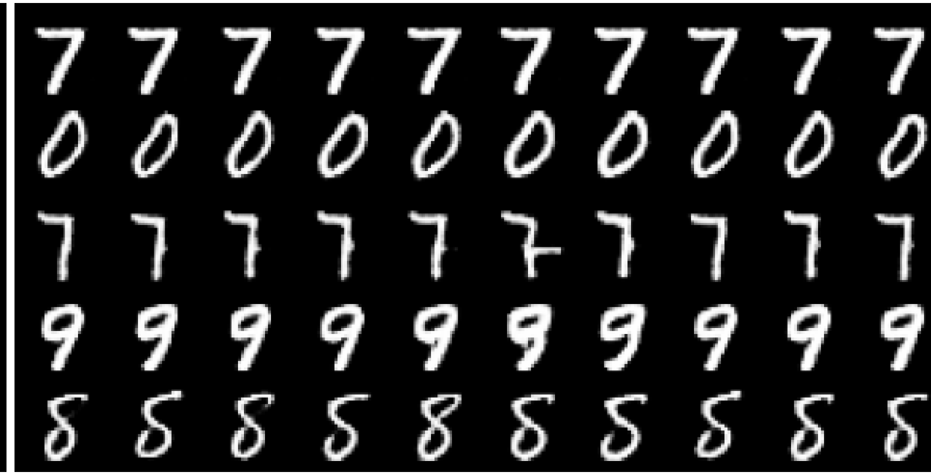


- c must have clear influence on x, so the encoder can recover c from x
- c will be easy to interpret

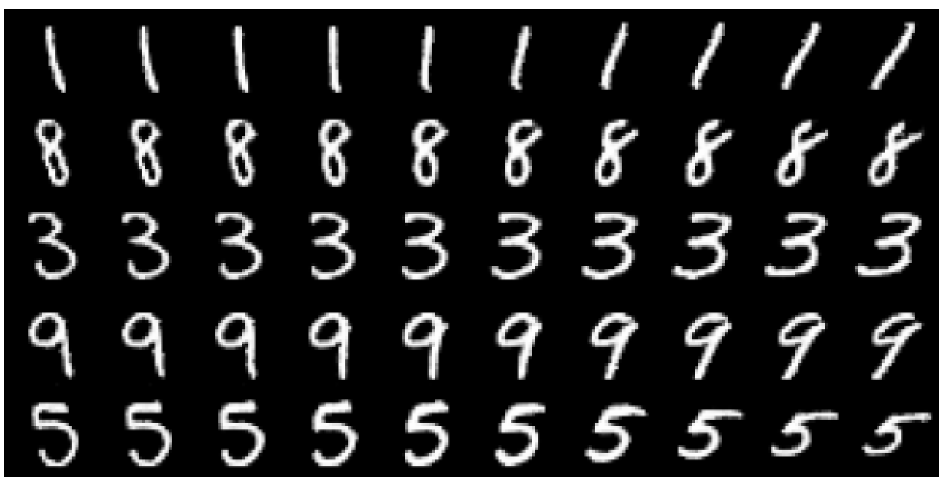
A specific dimension c_i cannot cooperate with other feature dimensions to have influence.



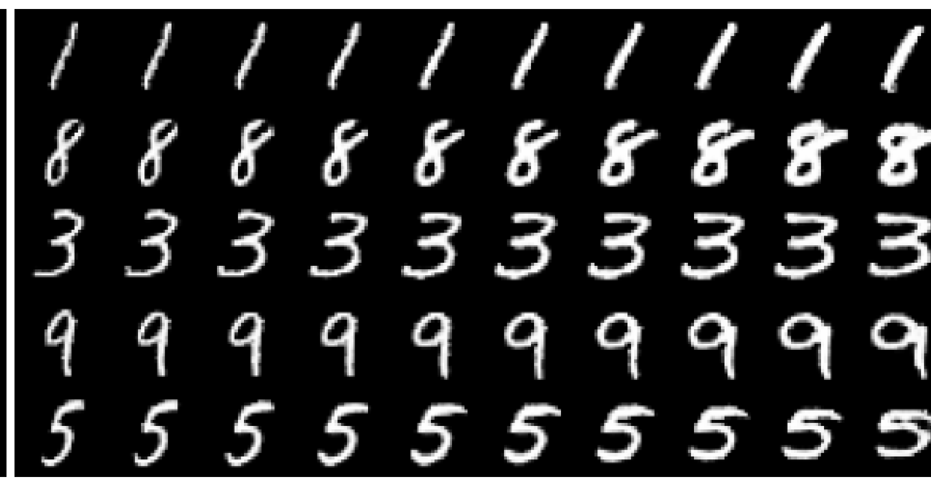
(a) Varying c_1 on InfoGAN (Digit type)



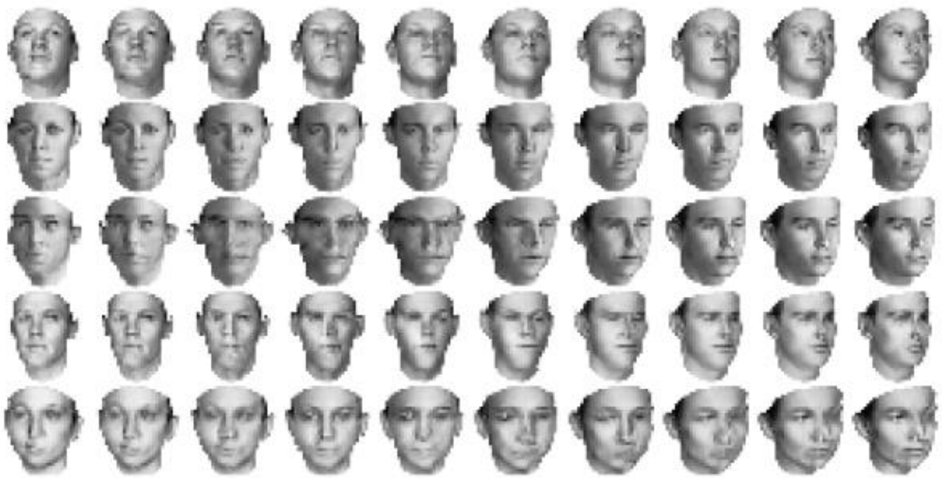
(b) Varying c_1 on regular GAN (No clear meaning)



(c) Varying c_2 from -2 to 2 on InfoGAN (Rotation)



(d) Varying c_3 from -2 to 2 on InfoGAN (Width)



(a) Azimuth (pose)

(b) Elevation

(c) Lighting

(d) Wide or Narrow

(a) Rotation (b) Width

(a) Continuous variation: Lighting (b) Discrete variation: Plate Context

(a) Azimuth (pose)

(b) Presence or absence of glasses

(c) Hair style

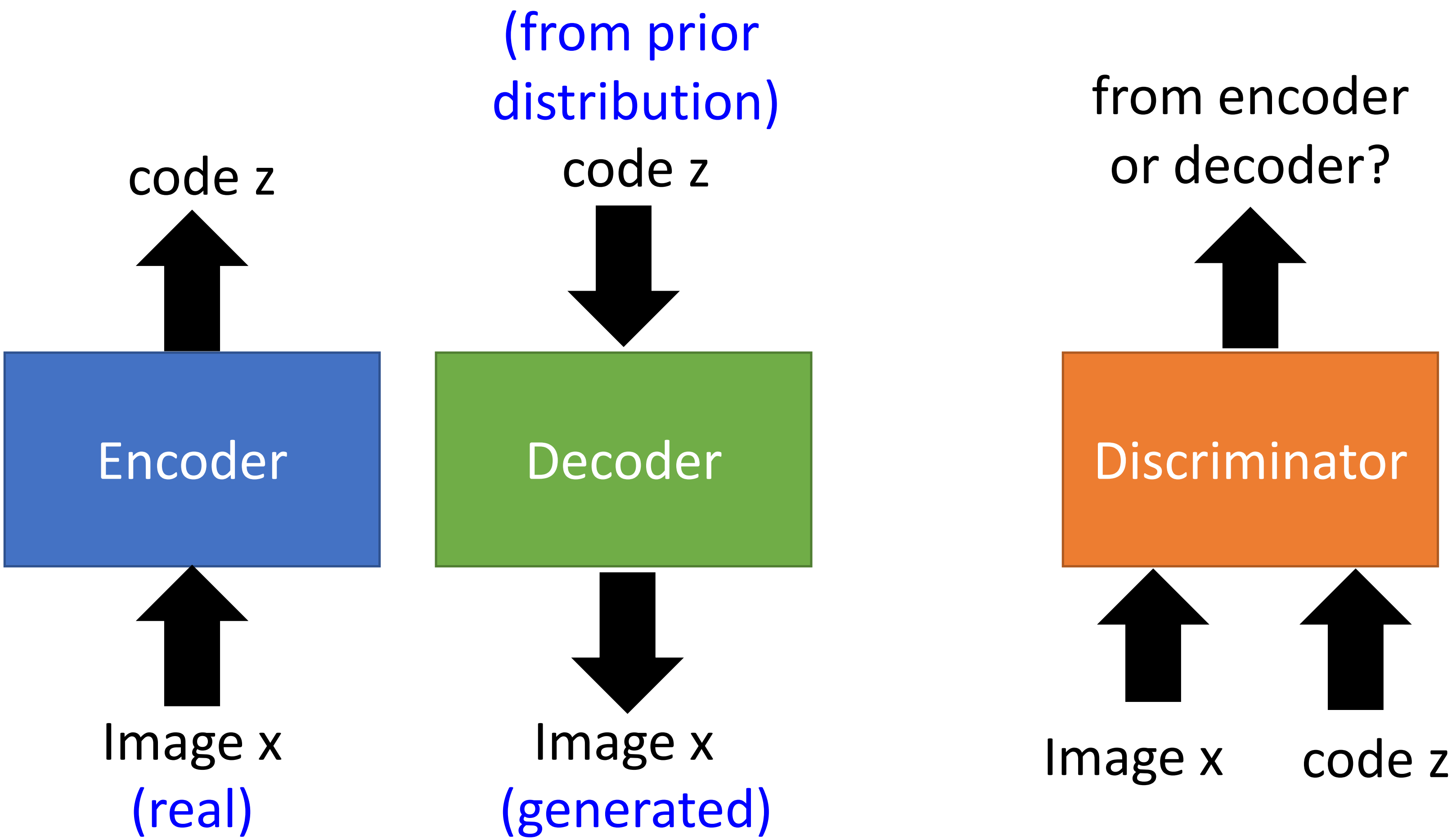
(d) Emotion

BiGAN

Jeff Donahue, Philipp Krähenbühl, Trevor Darrell, “Adversarial Feature Learning”, ICLR, 2017

Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, Aaron Courville, “Adversarially Learned Inference”, ICLR, 2017

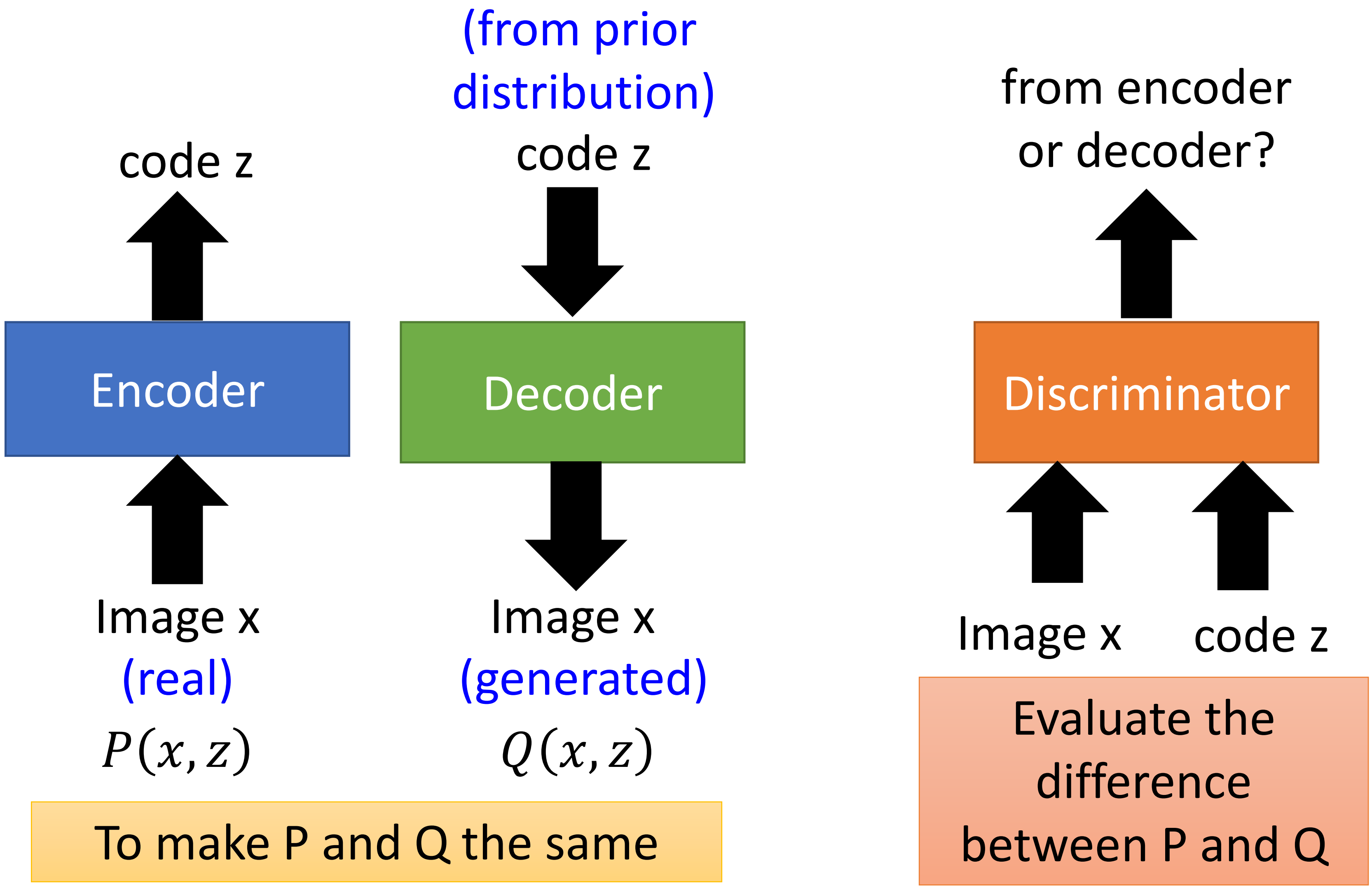
BiGAN



(from prior distribution)
code z
from encoder or decoder?
code z
Encoder
Decoder
Discriminator
Image x
(real)
Image x
(generated)
Image x
code z

Algorithm

- Initialize encoder En, decoder De, discriminator Dis
- In each iteration:
  - Sample M images $x^1, x^2, \cdots, x^M$ from database
  - Generate M codes $\tilde{z}^1, \tilde{z}^2, \cdots, \tilde{z}^M$ from encoder
    - $\tilde{z}^i = En\left(x^i\right)$
  - Sample M codes $z^1, z^2, \cdots, z^M$ from prior P(z)
  - Generate M codes $\tilde{x}^1, \tilde{x}^2, \cdots, \tilde{x}^M$ from decoder
    - $\tilde{x}^i = De\left(z^i\right)$
  - Update Dis to increase $Dis\left(x^i, \tilde{z}^i\right)$, decrease $Dis\left(\tilde{x}^i, z^i\right)$
  - Update En and De to decrease $Dis\left(x^i, \tilde{z}^i\right)$, increase $Dis\left(\tilde{x}^i, z^i\right)$



Optimal encoder and decoder:

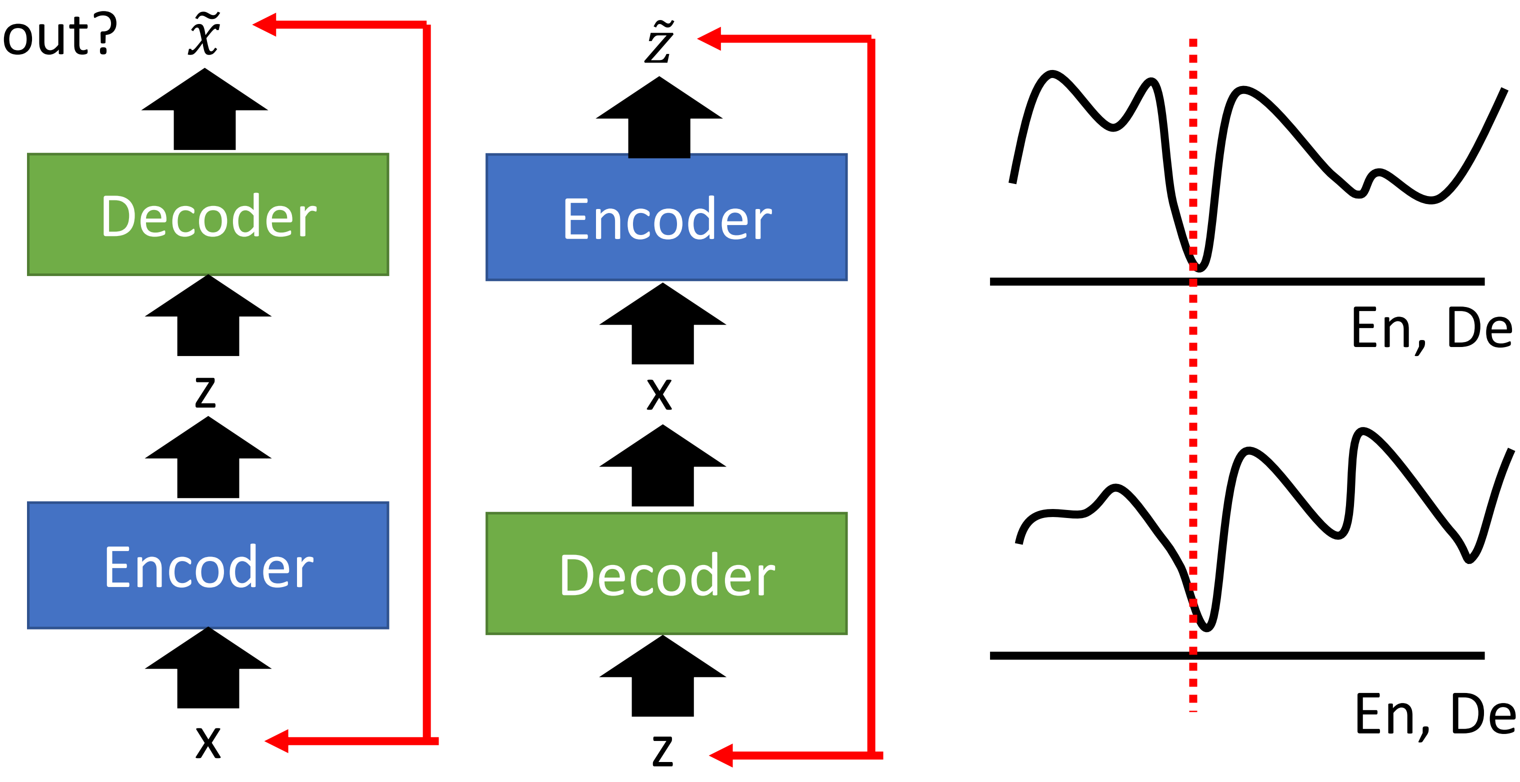
En(x') = z' $\Rightarrow$ De(z') = x' For all x'

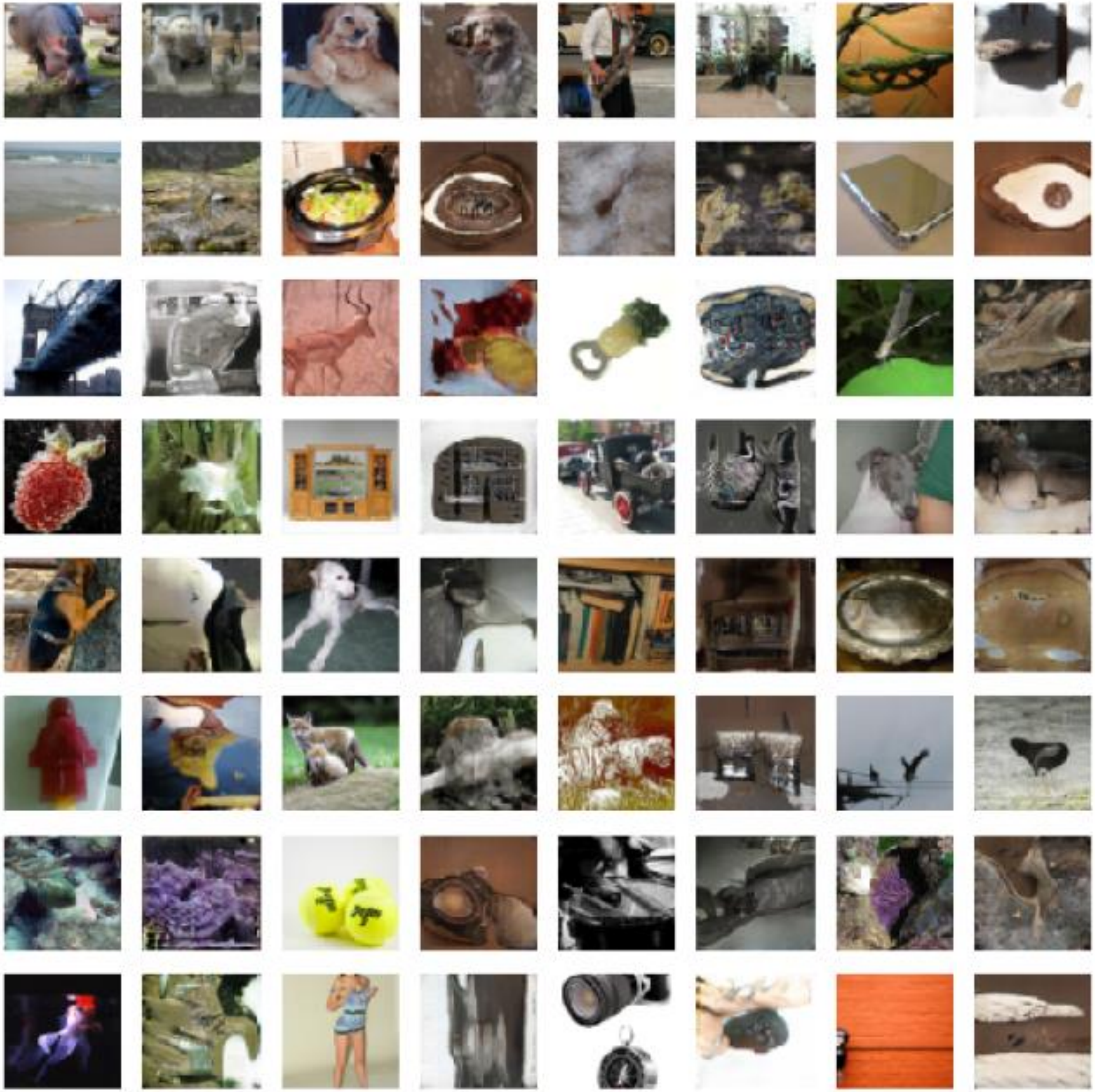
De(z'') = x'' $\Rightarrow$ En(x'') = z'' For all z''

BiGAN

Optimal encoder and decoder:	En(x') = z'	⇒	De(z') = x'	For all x'
	De(z'') = x''	⇒	En(x'') = z''	For all z''

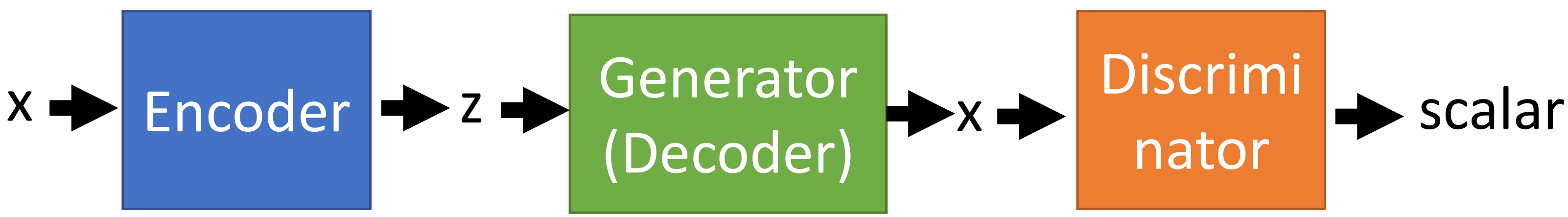
How about?





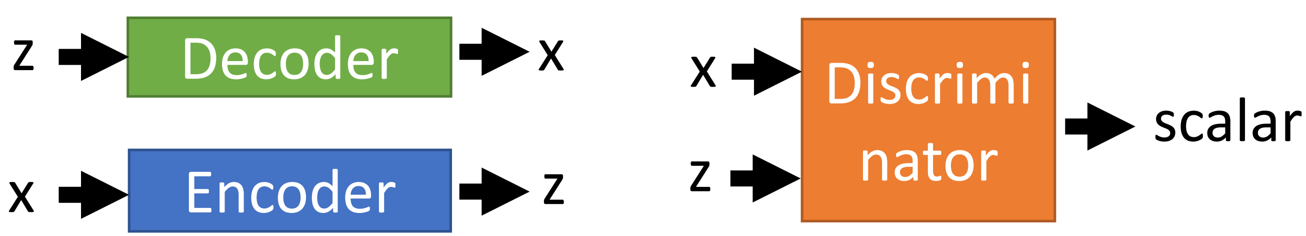
Concluding Remarks

- GAN+Autoencoder



- InfoGAN

- BiGAN



Next Time:
Energy-based GAN

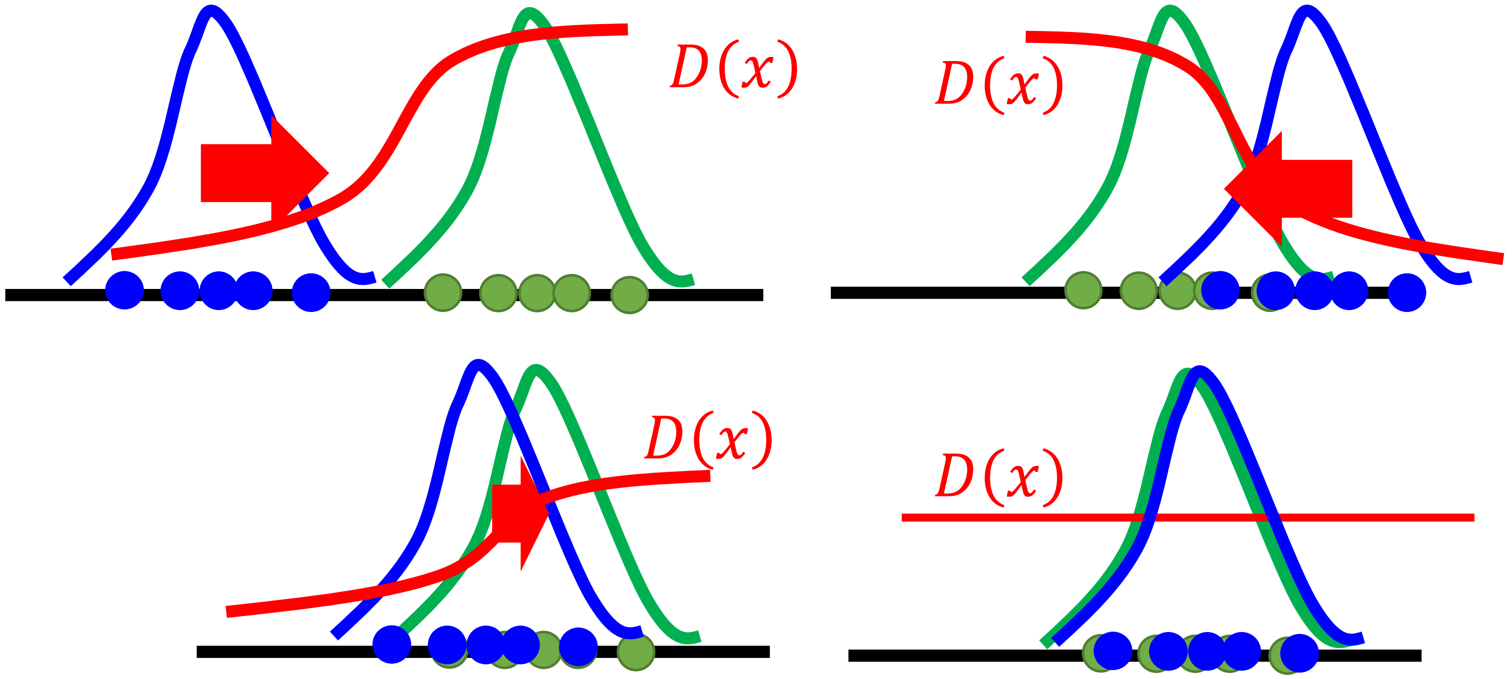
Original Idea

Discriminator

Data (target) distribution

Generated distribution

- Discriminator leads the generator

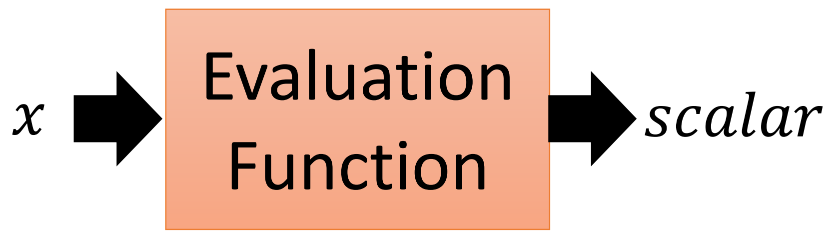


Original Idea

- When the data distribution and generated distribution is the same.
- The output of discriminator will be flat everywhere.
- However, discriminator is often used in pre-training.
  - It contains useful information.
- We always use the discriminator obtained in the last iteration as the initialization of the next step.

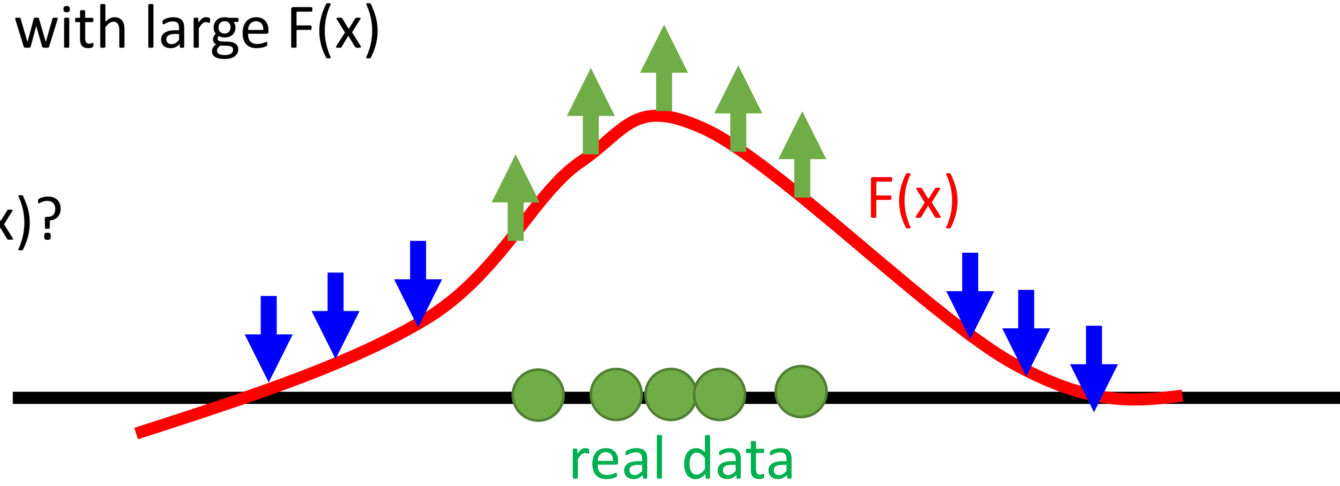
Energy-based Model

- We want to find an evaluation function F(x)
  - Input: object x (e.g. images), output: scalar (how good x is)
  - Real x has high F(x)
  - F(x) can be a network



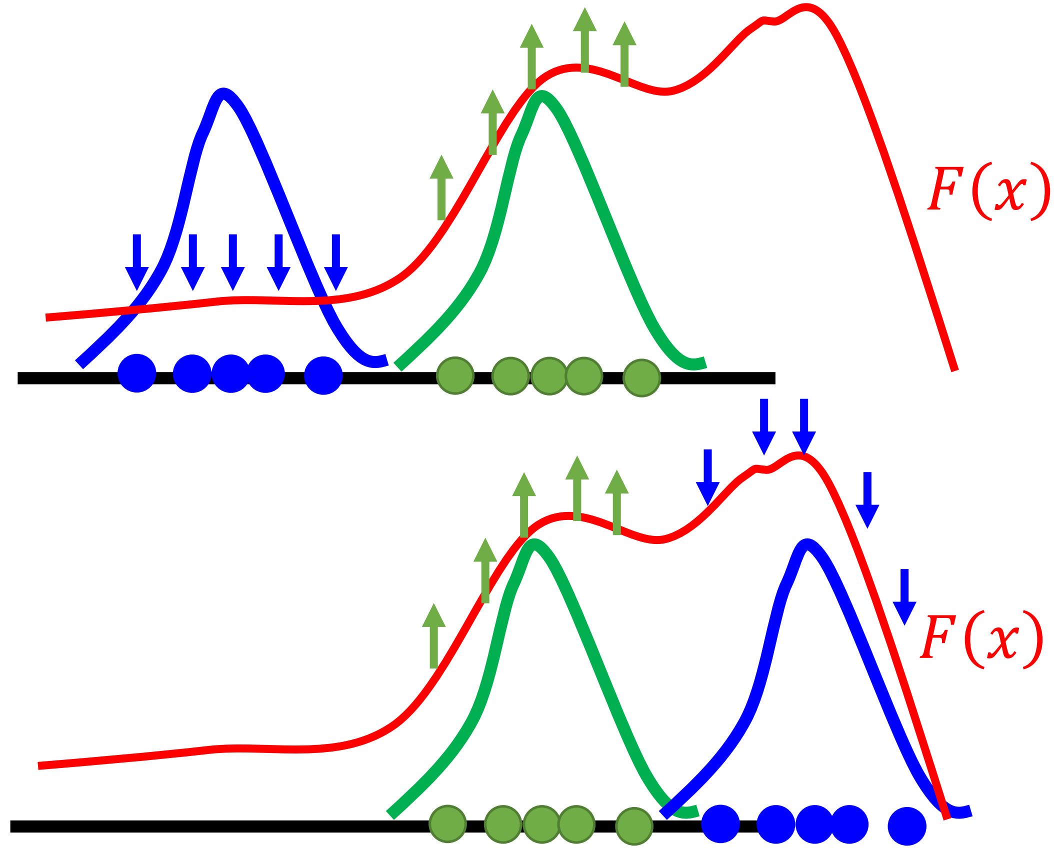
- We can find good x by F(x):
  - Generate x with large F(x)

- How to find F(x)?

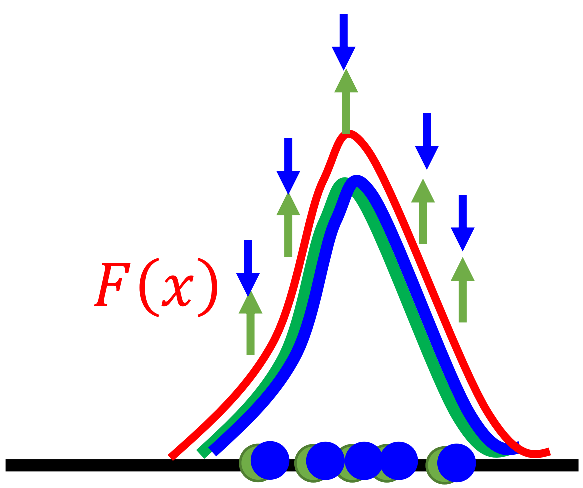


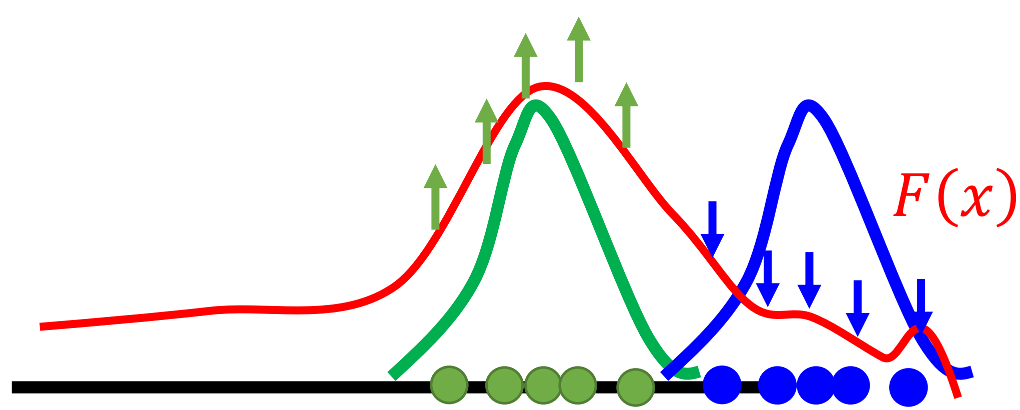
Energy-based GAN

- We want to find an evaluation function F(x)
- How to find F(x)?



In the end ……





Energy-based Model

- Preview: Framework of structured learning (Energy-based Model)
  - ML Lecture 21: Structured Learning - Introduction
    - https://www.youtube.com/watch?v=5OYu0vxXEv8
  - ML Lecture 22: Structured Learning - Linear Model
    - https://www.youtube.com/watch?v=HfPw40JPays
  - ML Lecture 23: Structured Learning - Structured SVM
    - https://www.youtube.com/watch?v=YjvGVVrCrhQ
  - ML Lecture 24: Structured Learning - Sequence Labeling
    - https://www.youtube.com/watch?v=o9FPSqobMys
  - Graphical model & Gibbs sampling
    - http://speech.ee.ntu.edu.tw/~tlkagk/courses/MLDS_2015_2/Lecture/MRF%20(v2).ecm.mp4/index.html