zoom pwd: 3Ewk6Fv!!

Deep Generative Models II

Generative Adversarial Networks

Machine Perception

Otmar Hilliges

30 April 2020

Advanced Interactive Technologies

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Last time

VAEs

Generative Adversarial Networks (Training objective)

This Week

Generative Adversarial Networks (GANs)

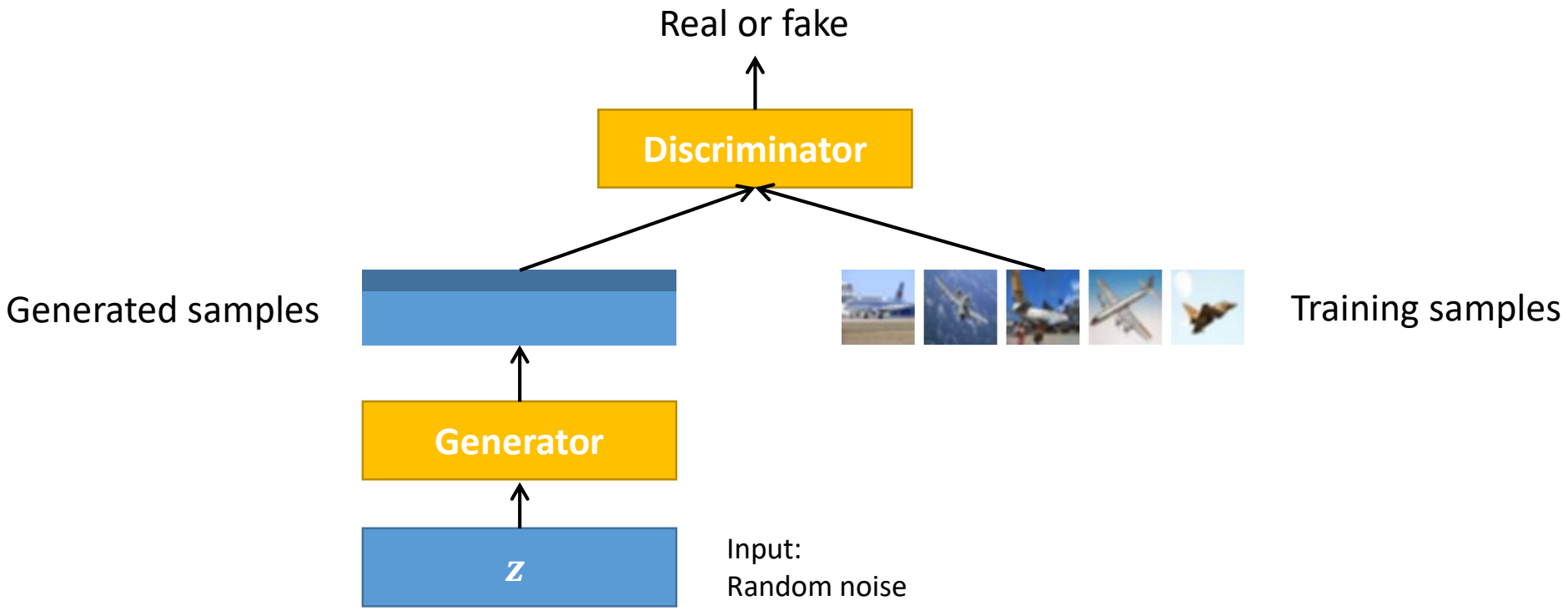
[Ian Goodfellow et al., “Generative Adversarial Nets”, NIPS 2014]

Training GANs: Intuition

General idea is to have two networks play a two-player game

Generator: try to fool the discriminator by generating real-looking images

Discriminator: try to distinguish between real and fake images



Training GANs: Objective – Part II

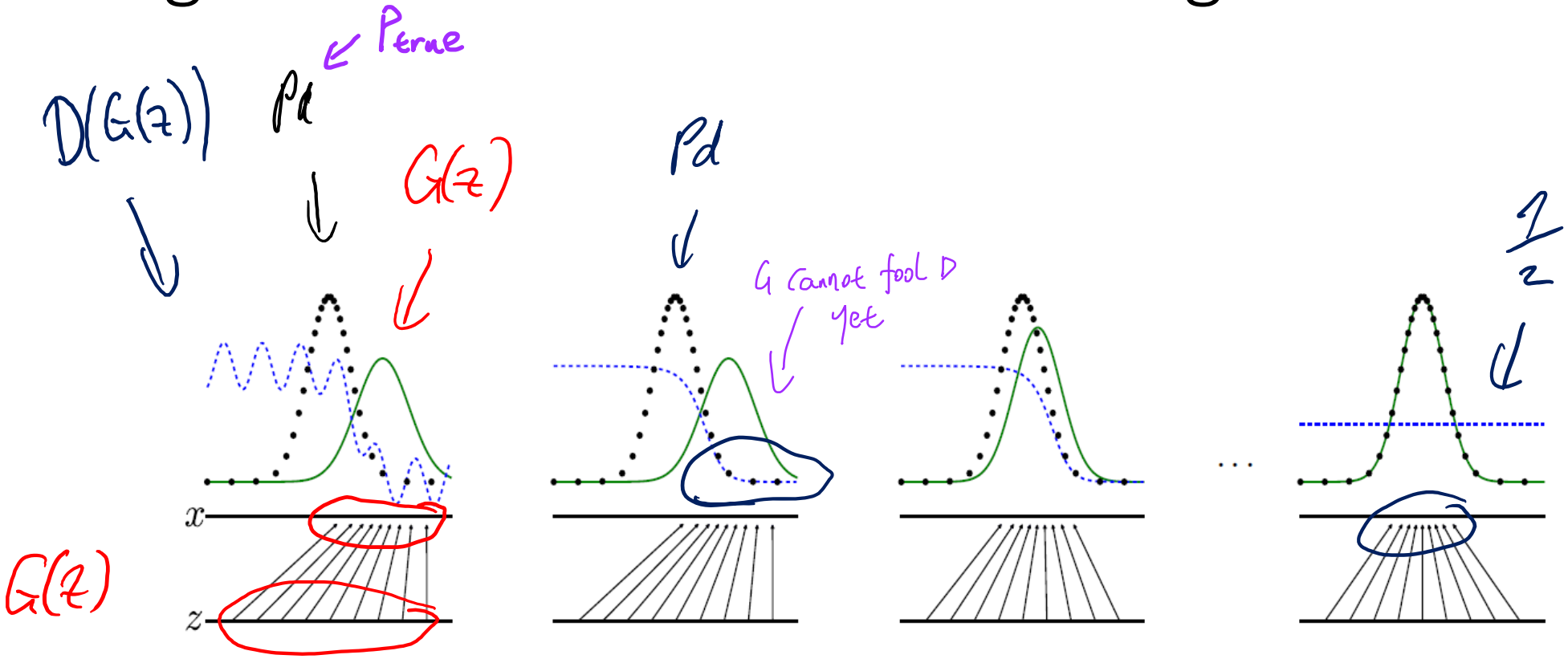
$$\arg\min_{G}\max_{D} V(G,D) \quad , \text{ where}$$

$$V(G,D) = \mathbb{E}_{x\sim p_d(x)}[\log(D(x)] + \mathbb{E}_{x\sim p_g(x)}[\log(1-D(x)]$$

$G(z)$

[Ian Goodfellow et al., “Generative Adversarial Nets”, NIPS 2014]

Training – Intuition and Idealized Progression



Training in practice - issue

Objective:

$$\min_{\Theta_g} \max_{\Theta_d} \mathbb{E}_{x\sim p_d(x)}\left[\log D_{\Theta_d}(x)\right] + \mathbb{E}_{z\sim p_z(z)}\left[\log\left(1 - D_{\Theta_d}\left(G_{\Theta_g}(z)\right)\right)\right]$$

Alternate between:

1. Gradient *ascent* on D:

$$\max_{\Theta_d} \mathbb{E}_{x\sim p_d(x)}\left[\log D_{\Theta_d}(x)\right] + \mathbb{E}_{z\sim p_z(z)}\left[\log\left(1 - D_{\Theta_d}\left(G_{\Theta_g}(z)\right)\right)\right]$$

2. Gradient *descent* on G:

$$\min_{\Theta_g} \mathbb{E}_{z\sim p_z(z)}\left[\log\left(1 - D_{\Theta_d}\left(G_{\Theta_g}(z)\right)\right)\right]$$

真假图片的数量要相等

Training algorithm (pseudo-code)

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**

  for k steps **do**

  - Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  - Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
  - Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right].$$

  end for

  - Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  - Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

end for

[Ian Goodfellow et al., "Generative Adversarial Nets", NIPS 2014]

Training in practice - issue

Objective:

$$\min_{\Theta_g} \max_{\Theta_d} \mathbb{E}_{x\sim p_d(x)}\left[\log D_{\Theta_d}(x)\right] + \mathbb{E}_{z\sim p_z(z)}\left[\log\left(1 - D_{\Theta_d}\left(G_{\Theta_g}(z)\right)\right)\right]$$

in this region, generator produces bad images;
and the same time, the gradient is very small, we might have a bit vanishing problem, the generator learns too slow.

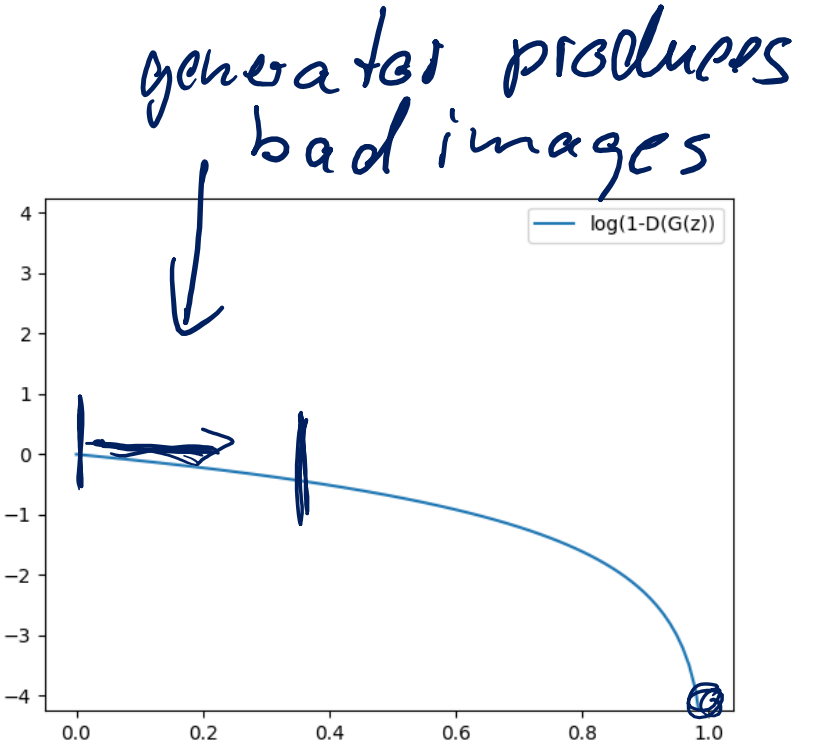
Alternate between:

1. Gradient *ascent* on D:

$$\max_{\Theta_d} \mathbb{E}_{x\sim p_d(x)}\left[\log D_{\Theta_d}(x)\right] + \mathbb{E}_{z\sim p_z(z)}\left[\log\left(1 - D_{\Theta_d}\left(G_{\Theta_g}(z)\right)\right)\right]$$

2. Gradient *descent* on G:

$$\min_{\Theta_g} \mathbb{E}_{z\sim p_z(z)}\left[\log\left(1 - D_{\Theta_d}\left(G_{\Theta_g}(z)\right)\right)\right]$$



Training in practice - fix

Objective:

$$\min_{\Theta_g} \max_{\Theta_d} \mathbb{E}_{x \sim p_d(x)} [\log D_{\Theta_d}(x)] + \mathbb{E}_{z \sim p_z(z)} \left[\log \left(1 - D_{\Theta_d} \left(G_{\Theta_g}(z) \right) \right) \right]$$

Alternate between:

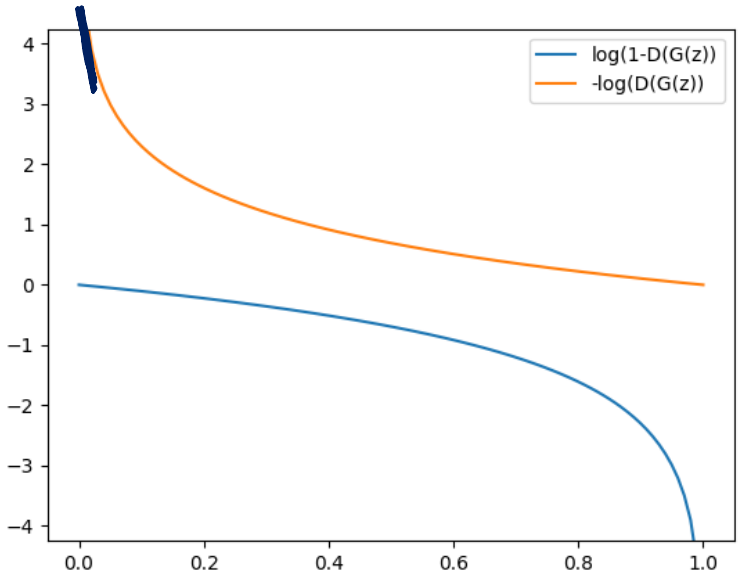
1. Gradient *ascent* on D

$$\max_{\Theta_d} \mathbb{E}_{x \sim p_d(x)} [\log D_{\Theta_d}(x)] + \mathbb{E}_{z \sim p_z(z)} \left[\log \left(1 - D_{\Theta_d} \left(G_{\Theta_g}(z) \right) \right) \right]$$

2. Instead: gradient *ascent* on G:

$$\max_{\Theta_g} \mathbb{E}_{z \sim p_z(z)} \left[\log \left(D_{\Theta_d} \left(G_{\Theta_g}(z) \right) \right) \right]$$

Steep grad when G is bad



Training algorithm (pseudo-code)

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**

  for k steps **do**

  - Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  - Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
  - Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right].$$

  end for

  - Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  - Update the generator by ascending its stochastic gradient (improved objective):

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(D_{\theta_d}\left(G_{\theta_g}\left(z^{(i)}\right)\right)\right)$$

end for

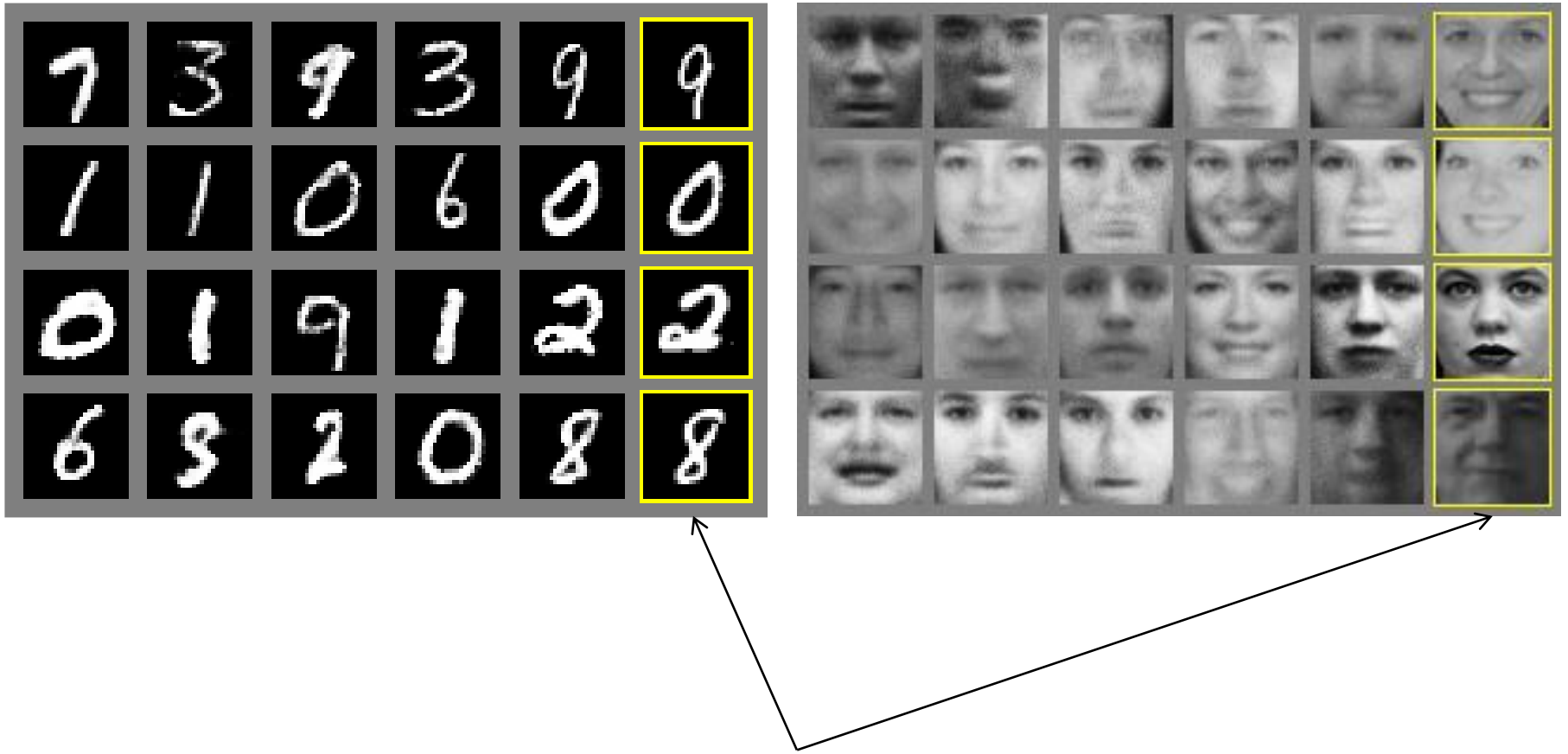
[Ian Goodfellow et al., "Generative Adversarial Nets", NIPS 2014]

before: descending

Update the generator by descending its stochastic gradient:

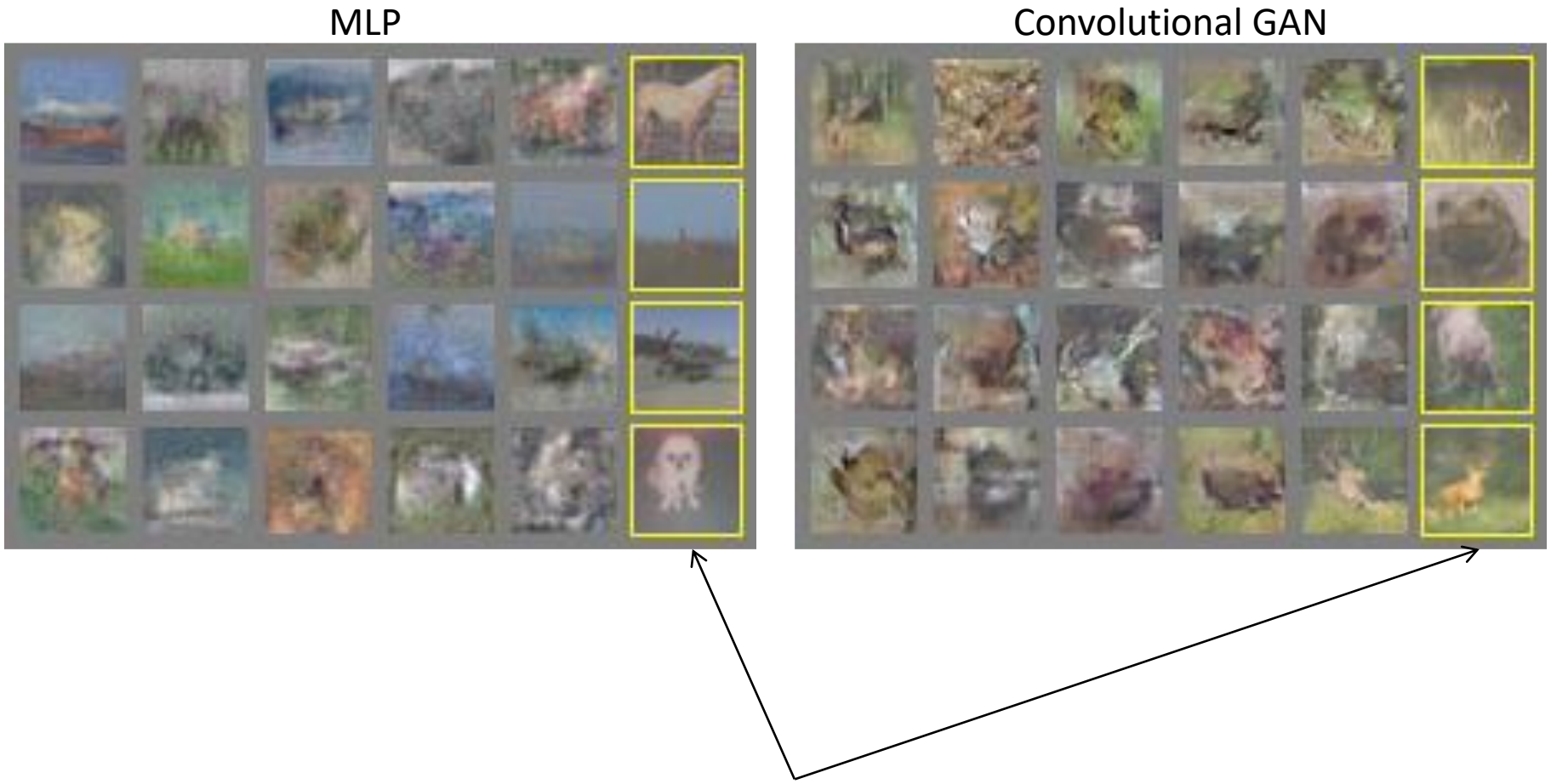
$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right)$$

GANs: Results MNIST & TFD



Nearest neighbors from training data

GANs: Results CIFAR-10



Issues

- Difficult to train:

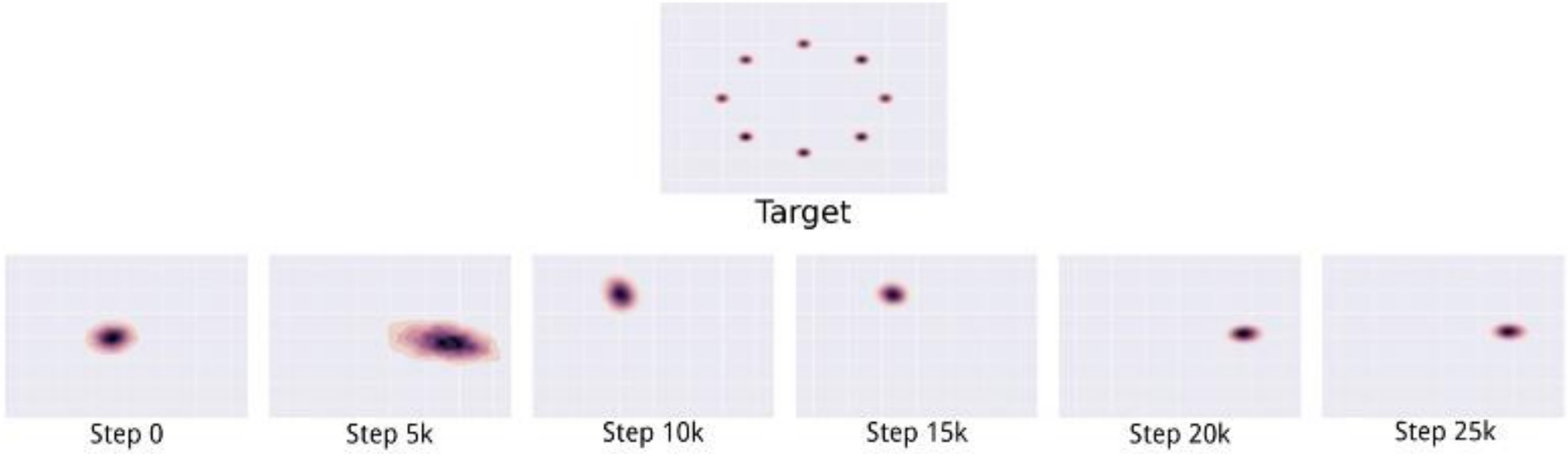
  Need to find Nash-Equilibrium in two-player game. Making downhill progress for one player may move the other player uphill.

  Generator can trick discriminator due to underlying image statistics

Issues

- Difficult to train:

  Mode collapse (no sample diversity): saddle point in dual energy landscape



[image source: Metz et al. Unrolled GANs. ICLR '17]

to fix this

Comparison with VAE

- Implicit pdf in VAE with intractable data likelihood
- No variational bound is needed (In GAN)
- GANs only care about samples
- Sharper images

- (Specific model families usable within GANs are already known to be universal approximators, so GANs are already known to be asymptotically consistent. Some VAEs are conjectured to be asymptotically consistent, but this is not yet proven)

Issues with Jensen–Shannon divergence => Wasserstein GAN

JSD correlates badly with sample quality → do not know when to stop training

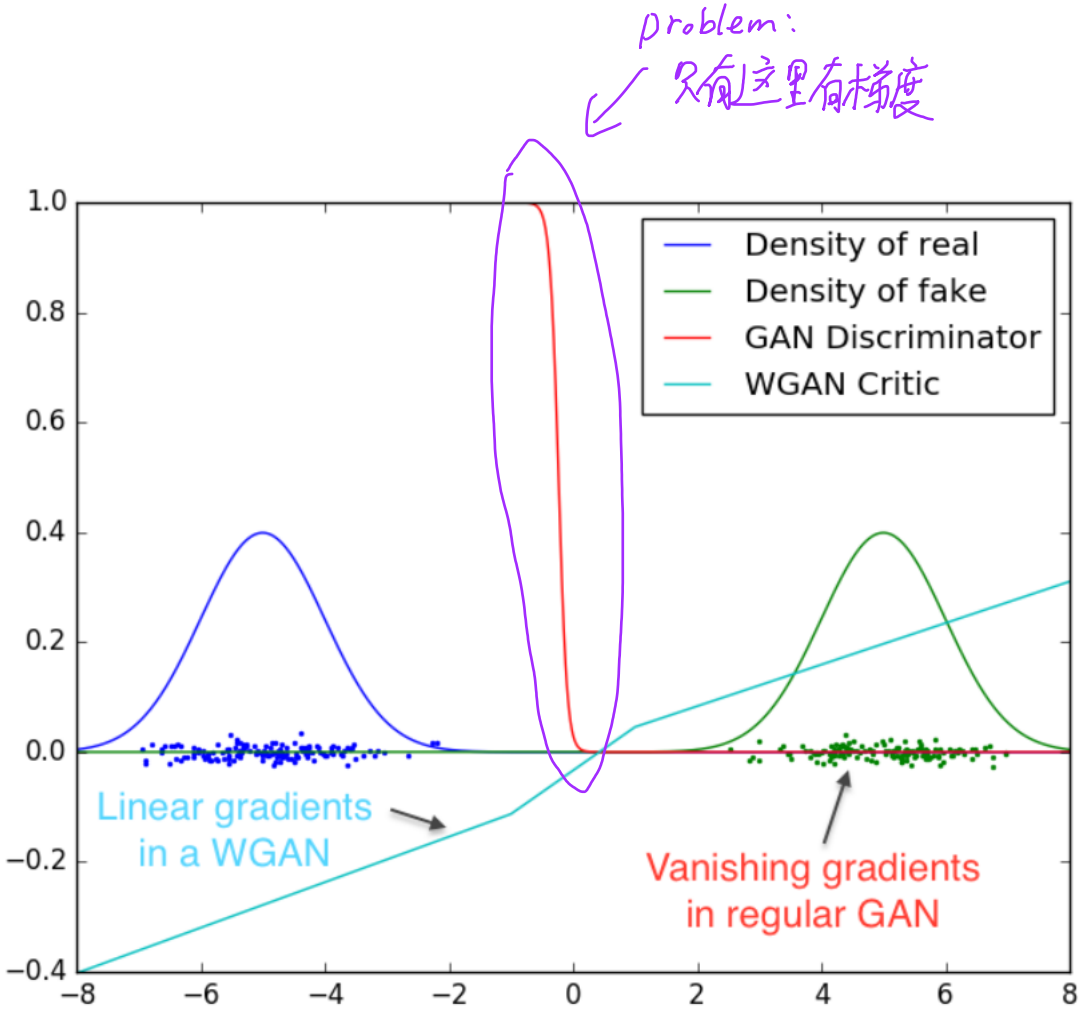
D tends to be optimal, JSD saturates → bad gradients

One solution: Use Wasserstein distance

earth mover's distance

GAN objective can be generalized to entire family of divergences:

[Nowozin et al. NeuIPS '16. f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization]



[image source: Arjovsky et al. Wasserstein GANs. '17]

GAN Extensions & Applications

Image-to-Image Translation with Conditional Adversarial Networks

[Isola et al. CVPR '17 – Pix2Pix]

pass sketch of img, translate it to real

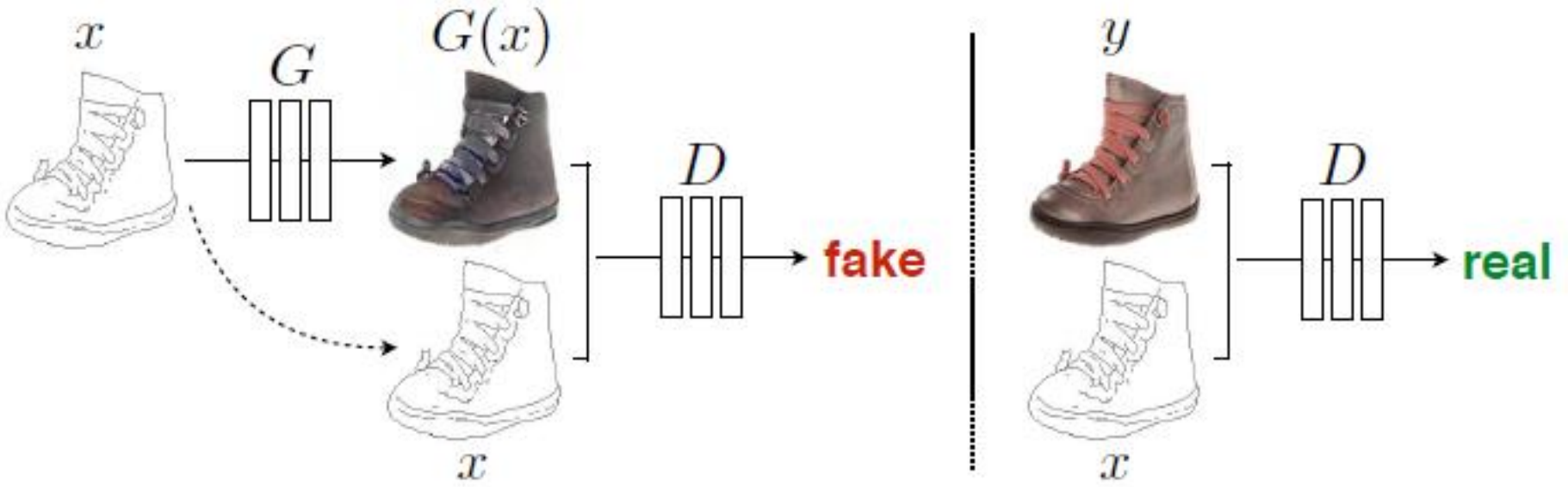


Image source: https://github.com/phillipi/pix2pix

[Isola et al. CVPR '17 – Pix2Pix]

Pix2Pix - Formulation

The Objective is:

$$\mathcal{L}(G, D) = \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G)$$

almost regular gan loss + reconstruction loss

where $\mathcal{L}_{cGAN}$ is a conditional GAN objective:

$$\mathcal{L}_{\text{cGAN}}(G, D) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_{x,z}[1 - \log D(x, G(x, z))]$$

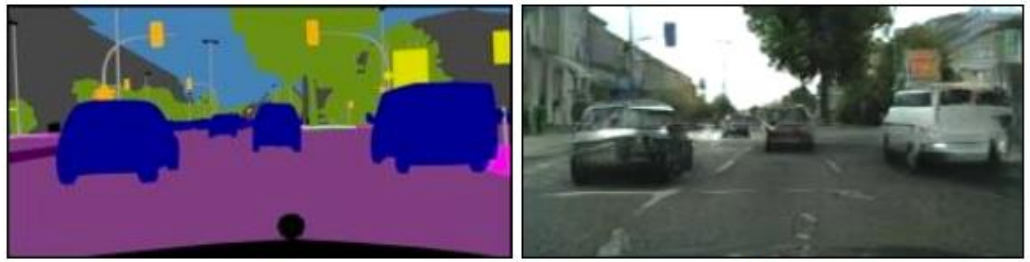
and $\mathcal{L}_{L1}$:

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}[\|y - G(x, z)\|_1]$$

Pix2Pix - results

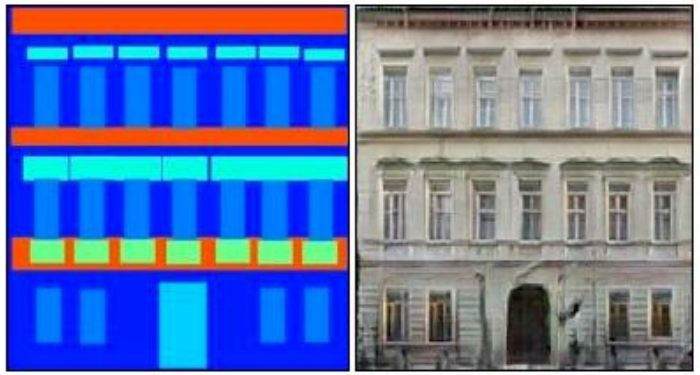
[Isola et al. CVPR ‘17 – Pix2Pix]

Labels to Street Scene



input output

Labels to Facade



input output

BW to Color

input output

Aerial to Map

input output

Day to Night

input output

Edges to Photo

input output

Pix2Pix - Limitations

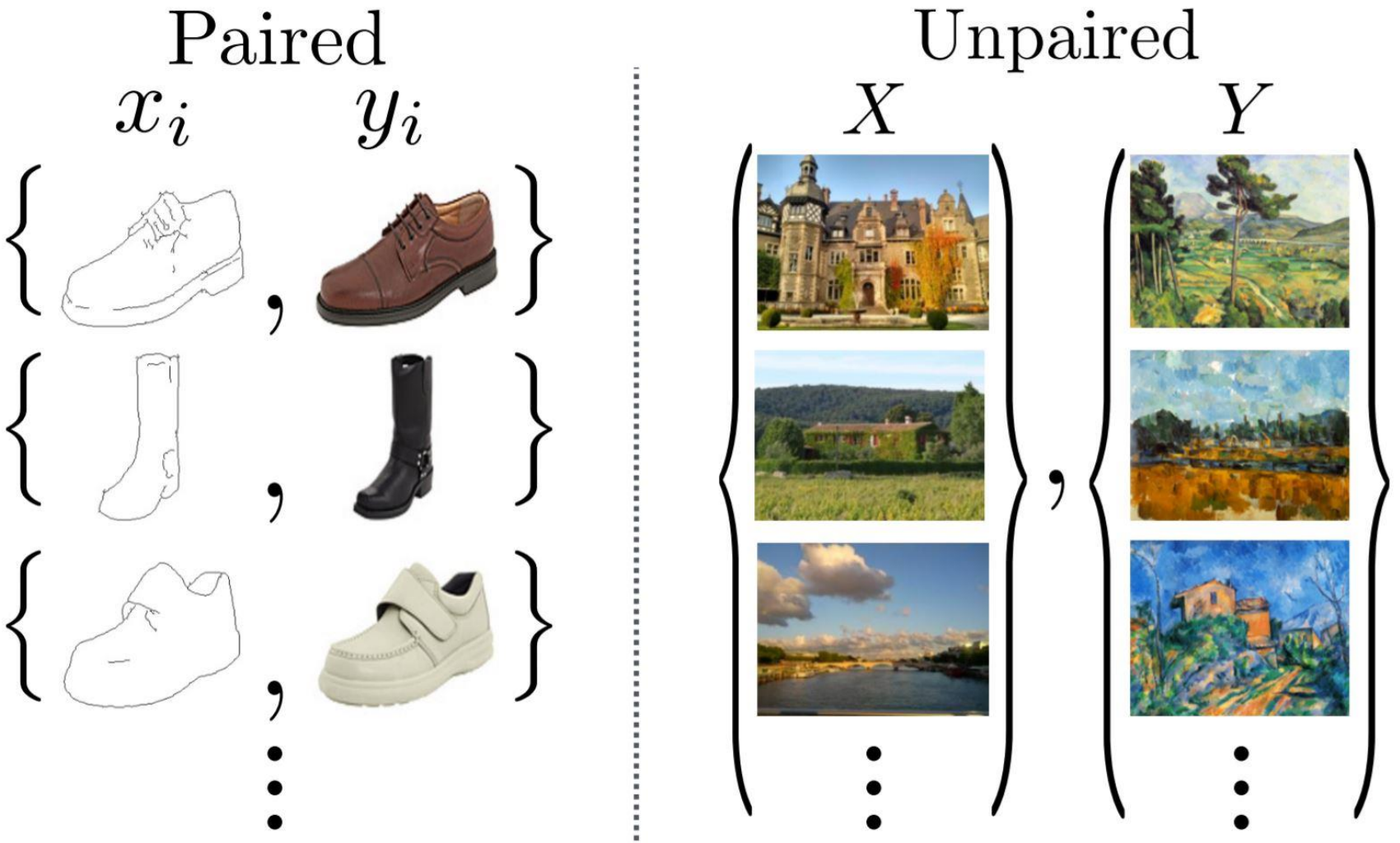
[Isola et al. CVPR ‘17 – Pix2Pix]

One obvious limitation is that this approach requires paired images as training data.

In many cases, one has at its disposal examples from two densities and wants to translate a sample from the first (“images of apples”) into a sample likely under the second (“images of oranges”).

CycleGAN – unpaired image translation

[Zhu et al. ICCV '17 – CycleGAN]



CycleGAN – unpaired image translation

[Zhu et al. ICCV '17 – CycleGAN]

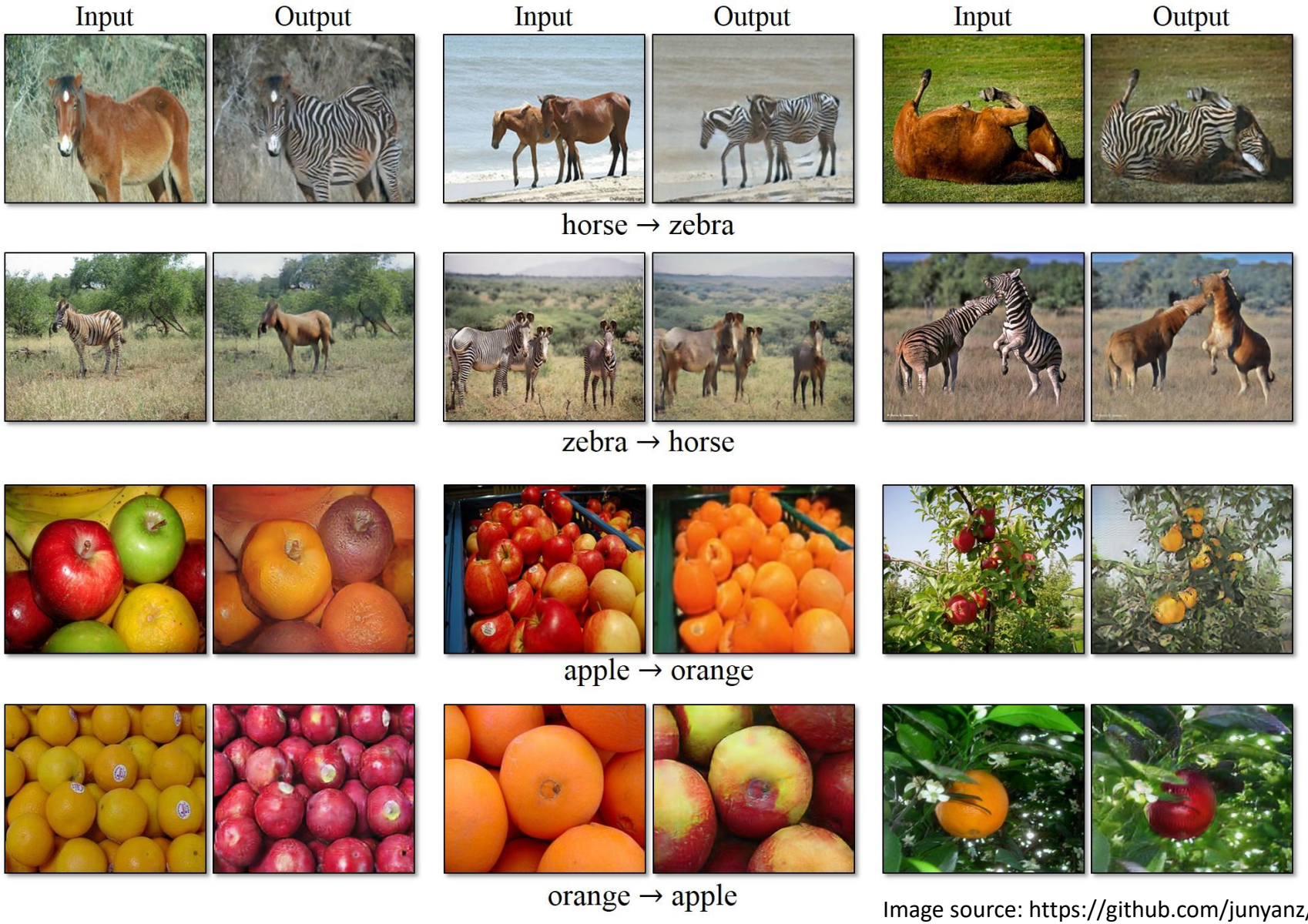
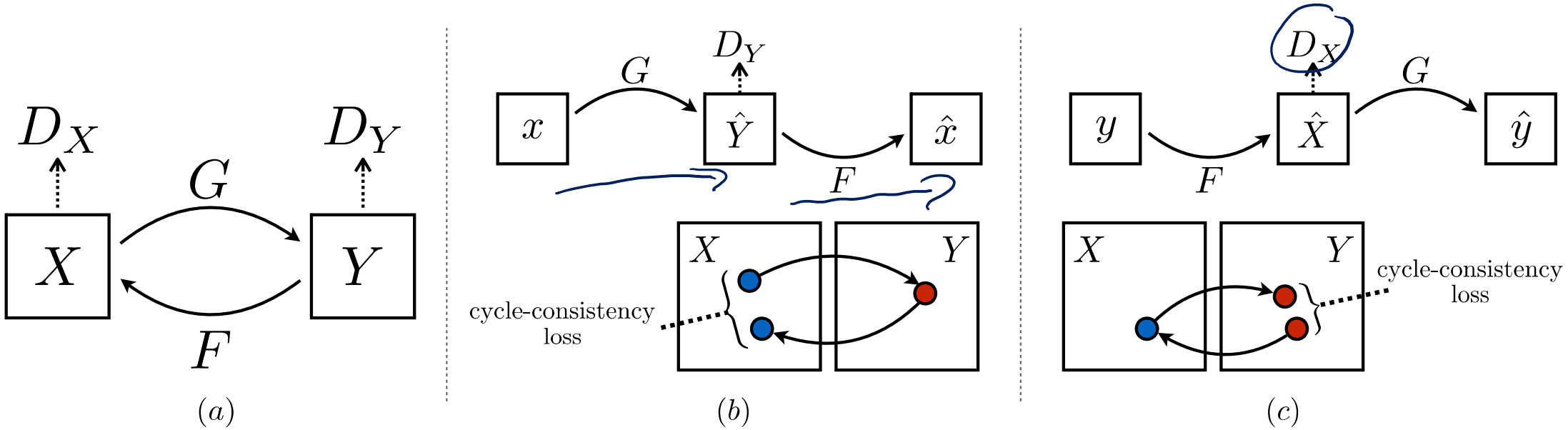


Image source: https://github.com/junyanz/CycleGAN/

CycleGAN - Overview

[Zhu et al. ICCV '17 – CycleGAN]



[Zhu et al. ICCV '17 – CycleGAN]

CycleGAN - Formulation

The Objective is:

$$\mathcal{L}(G, F, D_X, D_Y)$$
$$= \mathcal{L}_{GAN}(G, D_Y, X, Y) + \mathcal{L}_{GAN}(F, D_X, Y, X) + \lambda \mathcal{L}_{\text{cyc}}(G, F)$$

where $\mathcal{L}_{GAN}$ is the standard GAN loss as previously and $\mathcal{L}_{\text{cyc}}$ is:

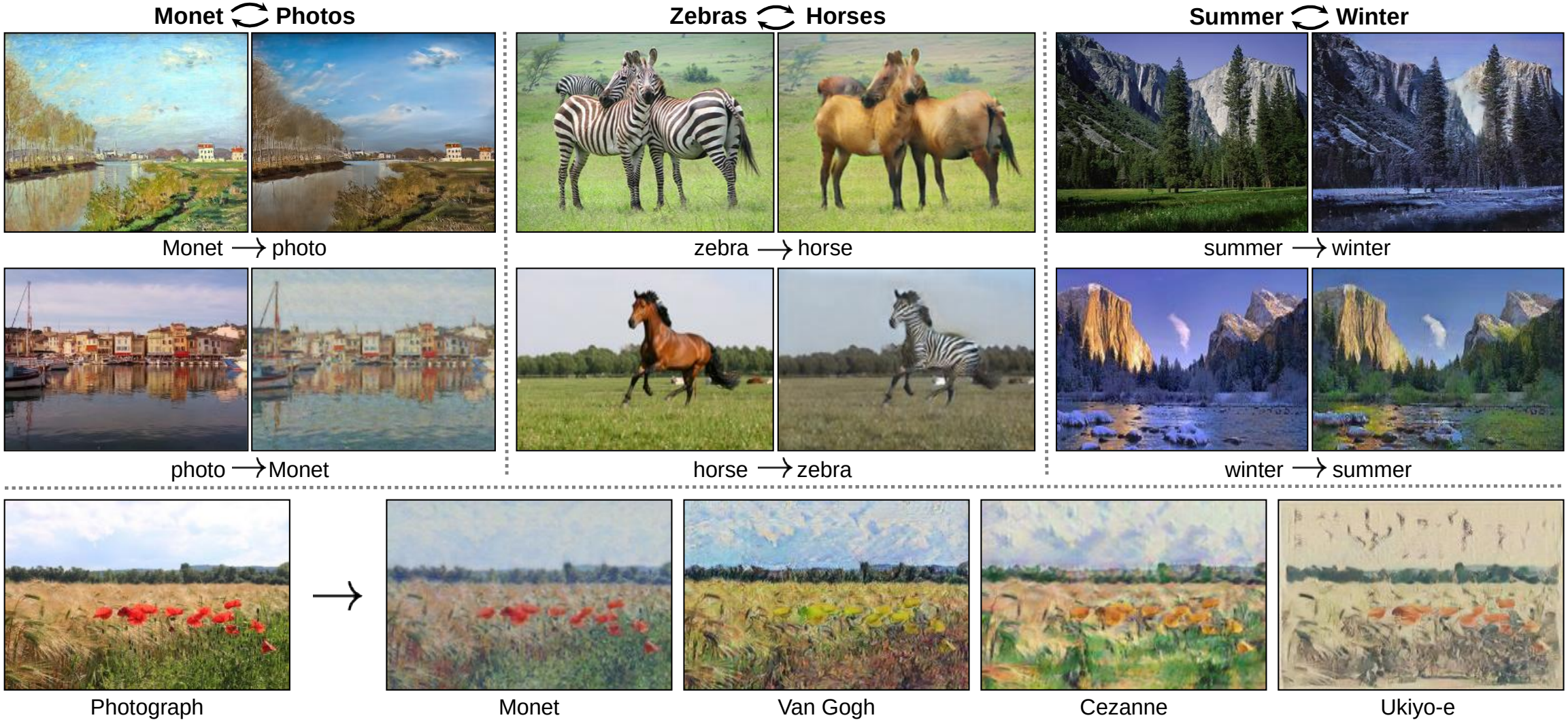
$$\mathcal{L}_{\text{cyc}}$$
$$= \mathbb{E}_{x \sim p_{data}(x)}[\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim p_{data}(y)}[\|G(F(y)) - y\|_1]$$

CycleGAN – Effect of cycle consistency

[Zhu et al. ICCV '17 – CycleGAN]

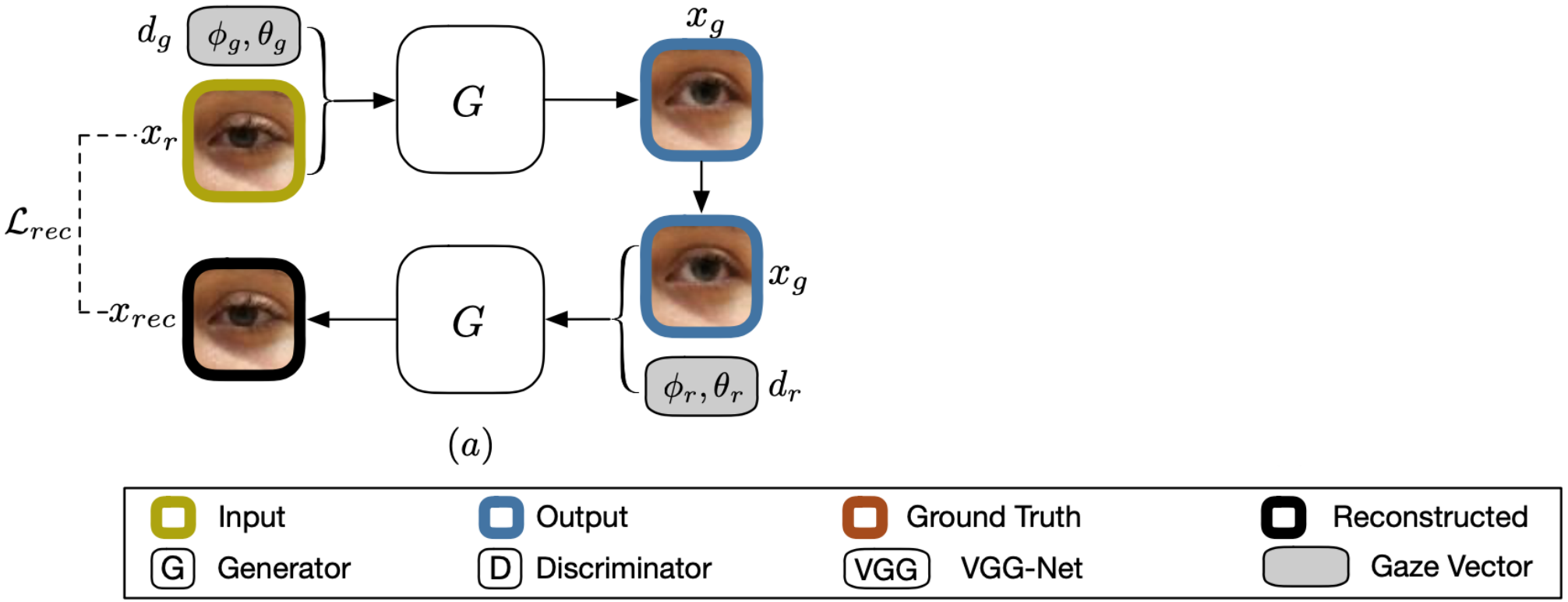
CycleGAN – More results

[Zhu et al. ICCV '17 – CycleGAN]



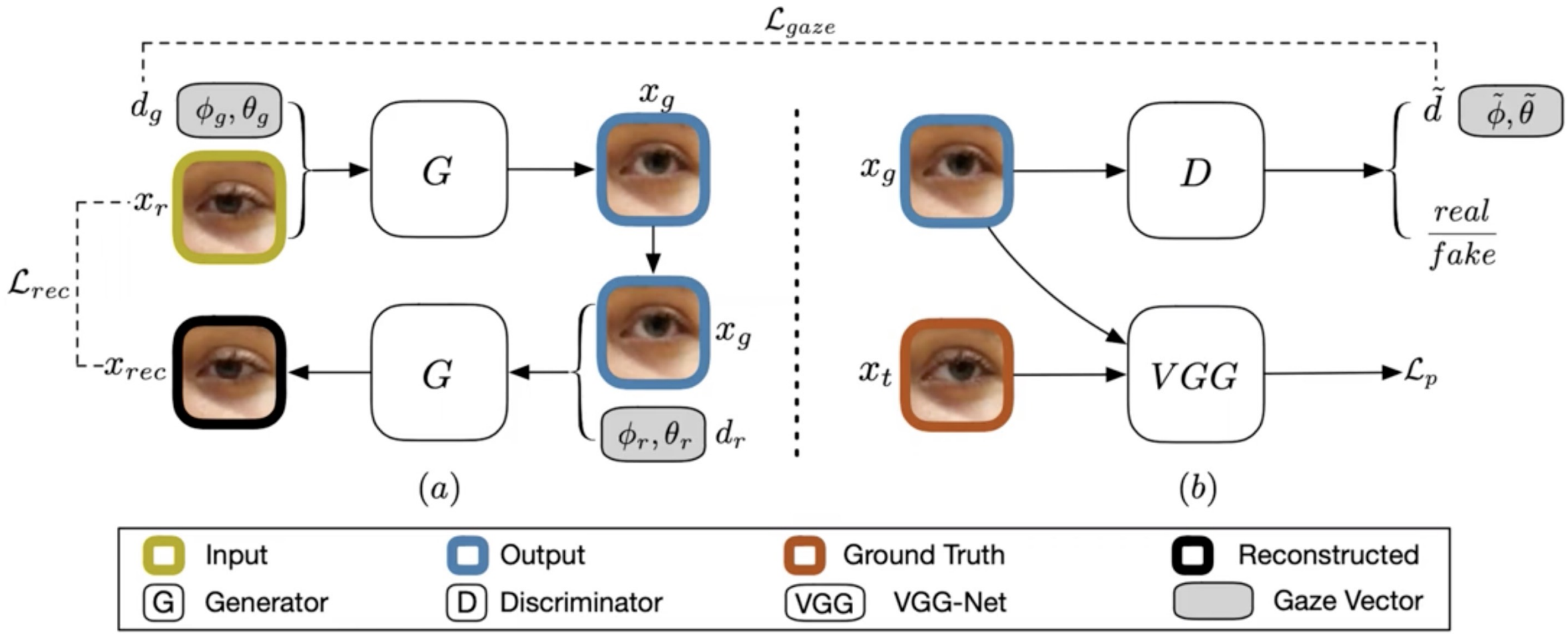
Code: https://github.com/HzDmS/gaze_redirection

Gaze redirection



[He et al. ICCV'19 Photo-realistic Gaze Redirection]

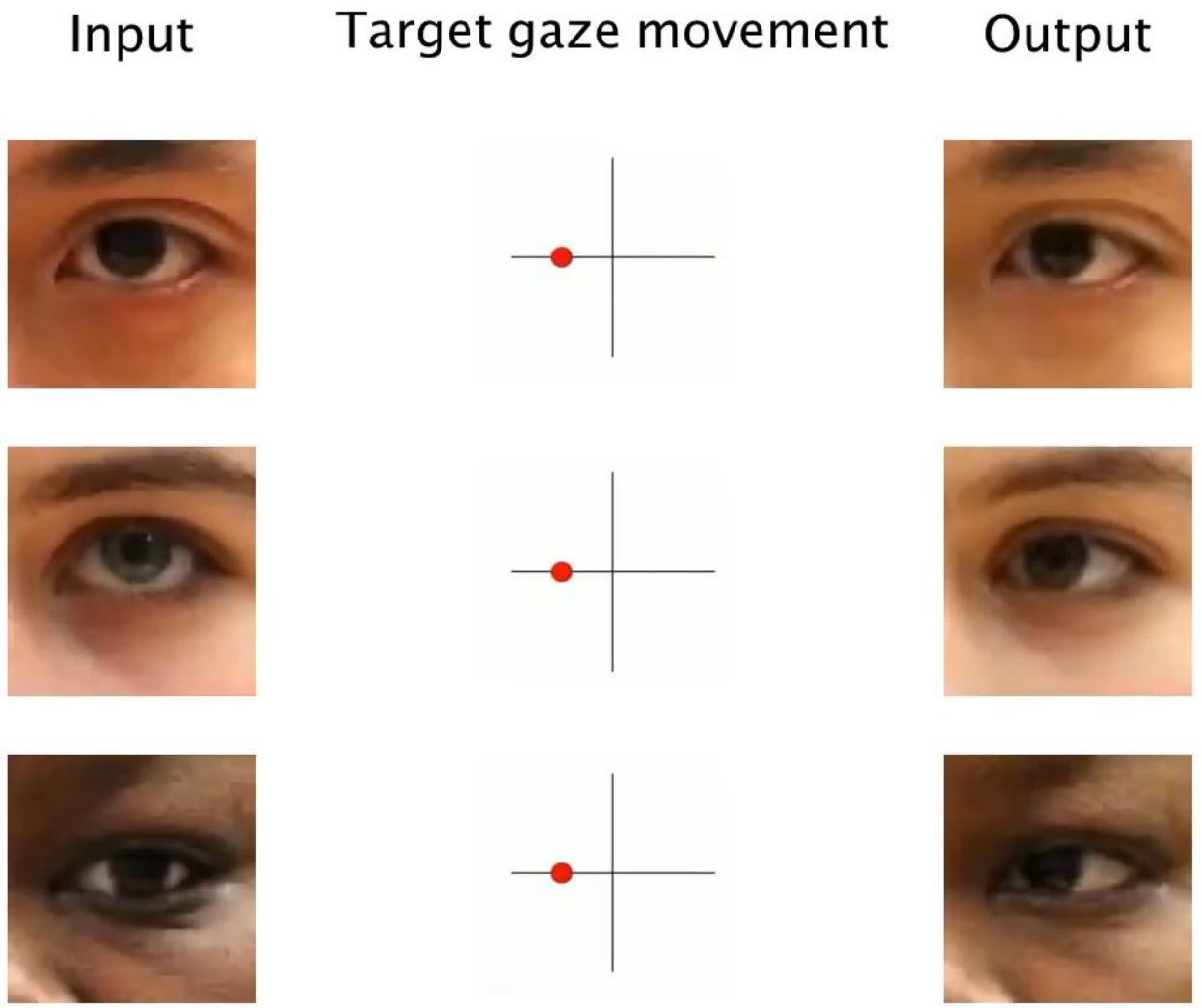
Gaze redirection



[He et al. ICCV'19 Photo-realistic Gaze Redirection]

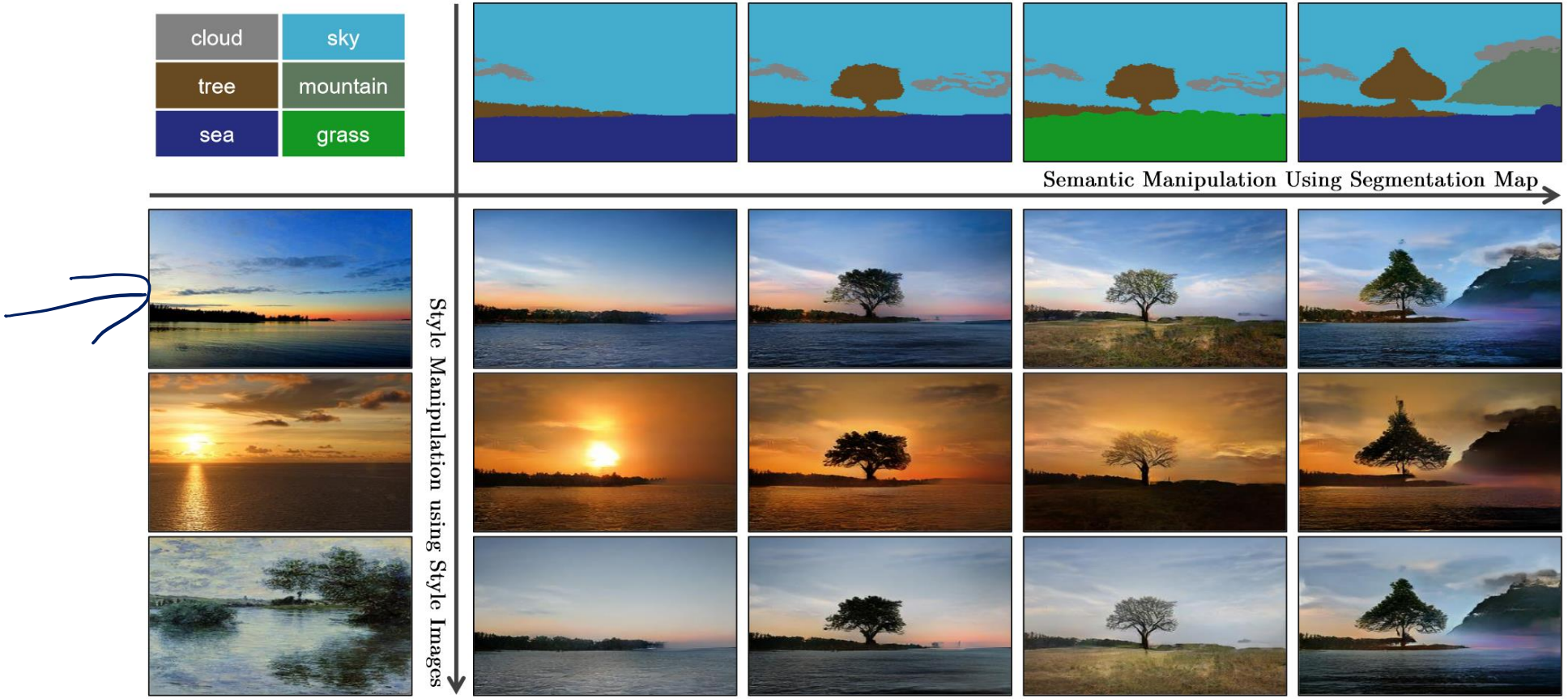
Demo

Code: https://github.com/HzDmS/gaze_redirection



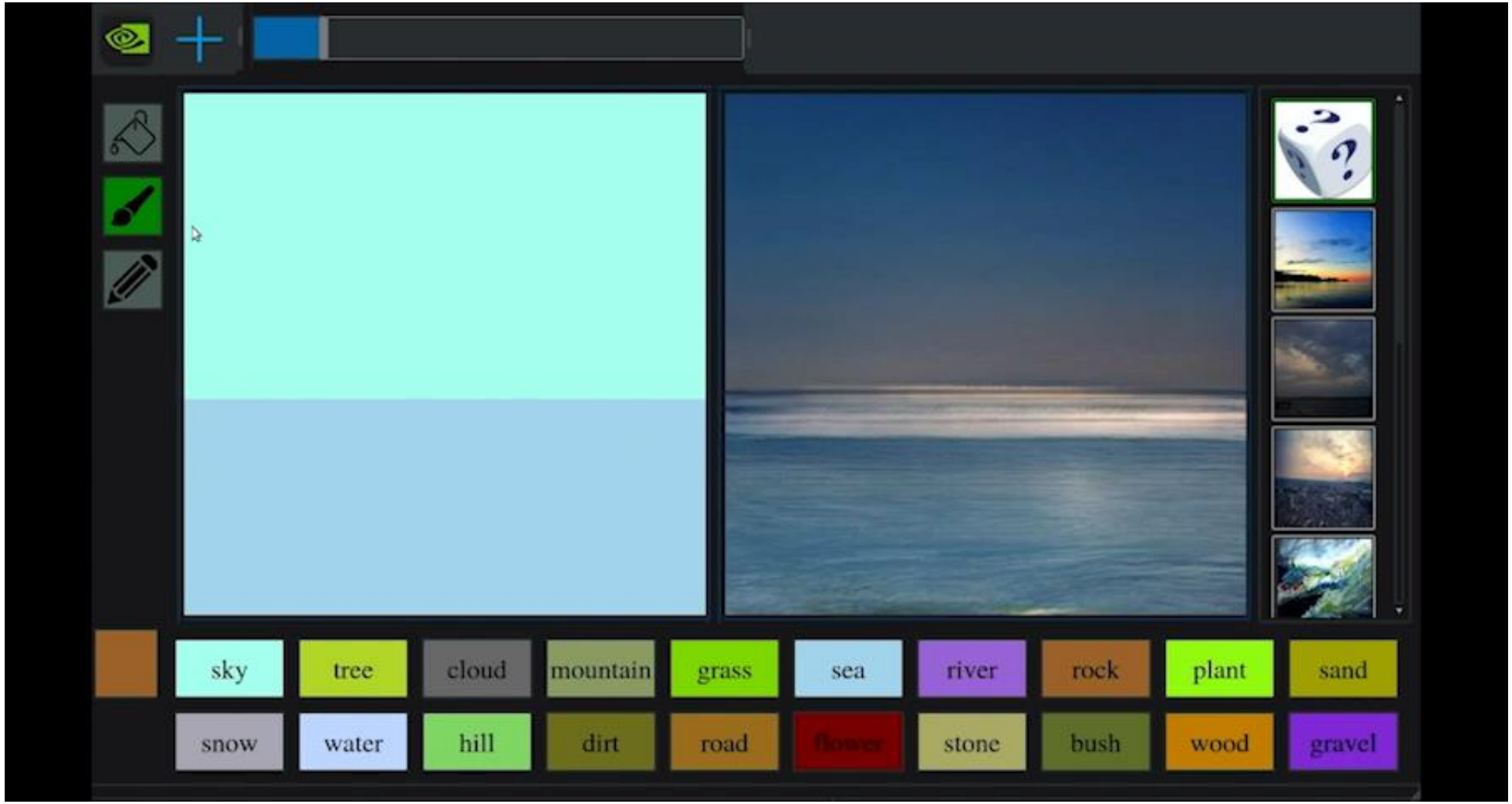
GauGAN - Semantic Image Synthesis with Spatially-Adaptive Normalization

Task: Given semantic segmentation and reference style, synthesize image



[Park et al. CVPR '19 - Semantic image synthesis with spatially-adaptive normalization]

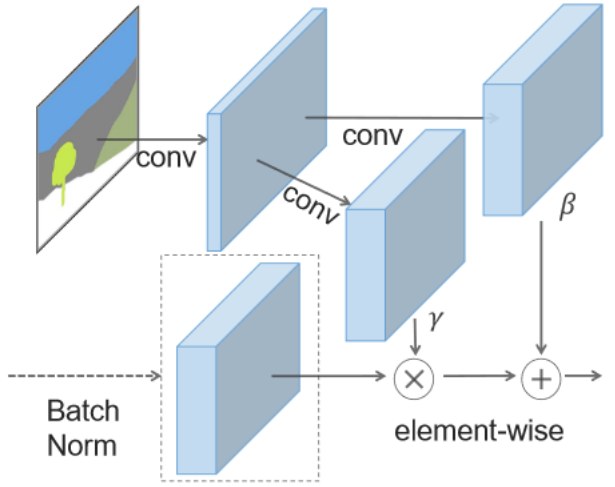
GauGAN - Semantic Image Synthesis with Spatially-Adaptive Normalization

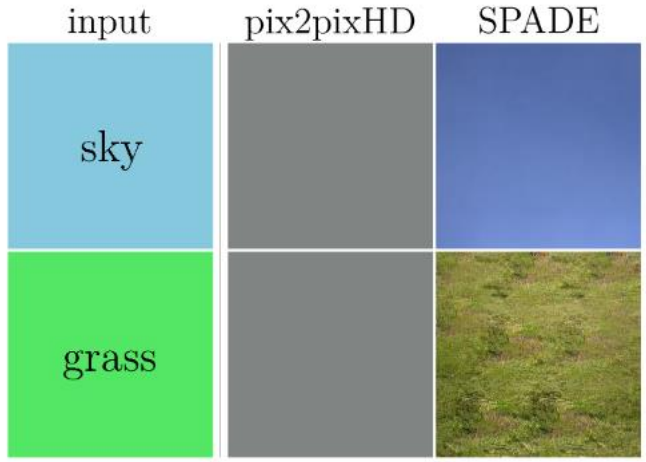


[Park et al. CVPR '19 - Semantic image synthesis with spatially-adaptive normalization]

Problem with pix2pix:

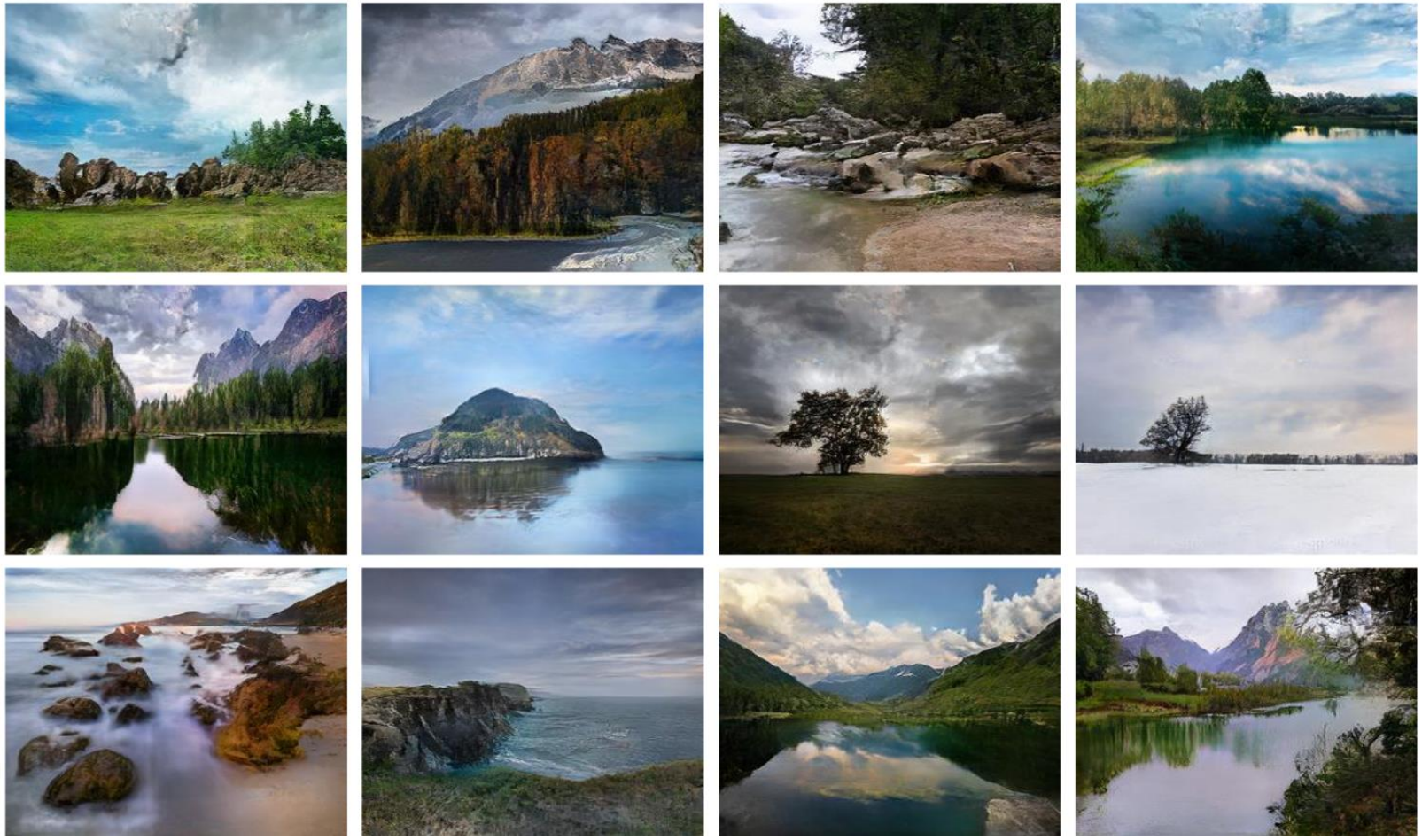
- Unconditional normalization layer in generator “washes away” information of semantic labels.
- Proposal: Spatially-varying conditional normalization





[Park et al. CVPR ‘19 - Semantic image synthesis with spatially-adaptive normalization]

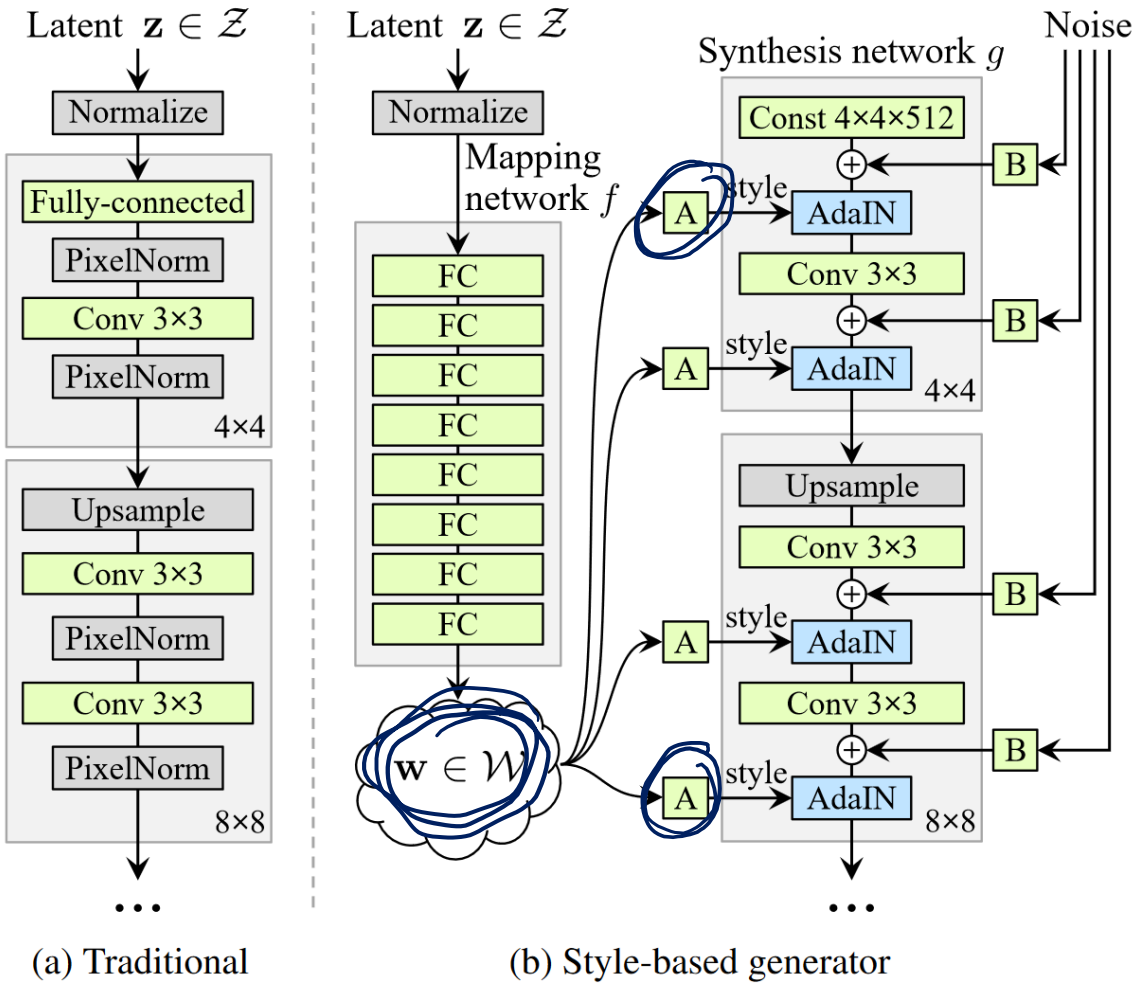
Results on Flickr Landscapes dataset



Try yourself: http://nvidia-research-mingyuliu.com/gaugan

[Park et al. CVPR ‘19 - Semantic image synthesis with spatially-adaptive normalization]

A Style-Based Generator Architecture for Generative Adversarial Networks



(a) Traditional (b) Style-based generator

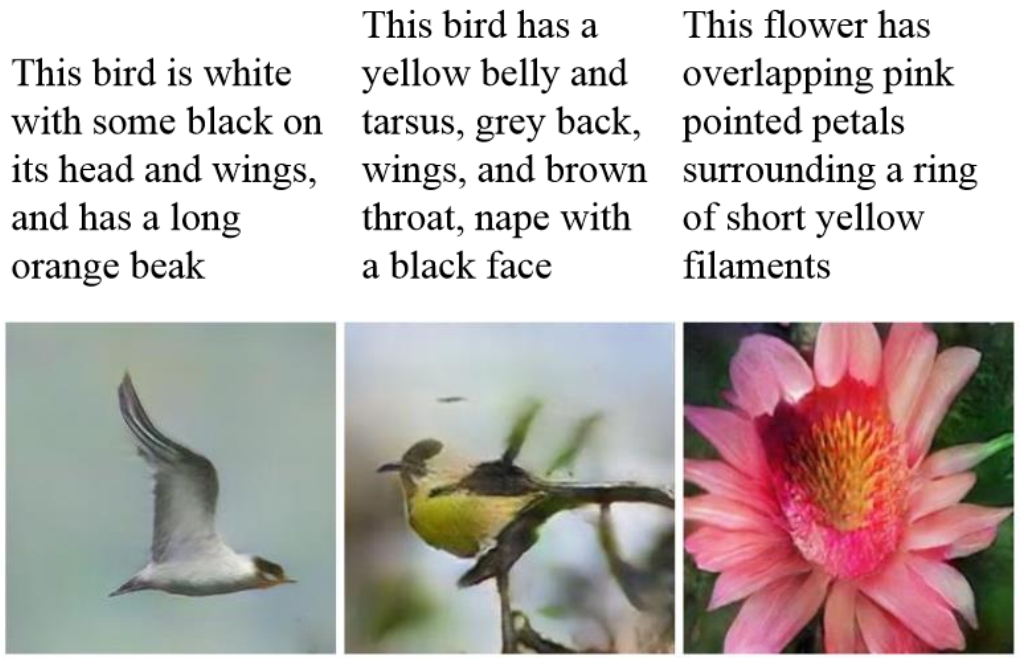
[Karras et al. CVPR '19 - A Style-Based Generator Architecture for Generative Adversarial Networks]

High-quality image generation

Granular style control

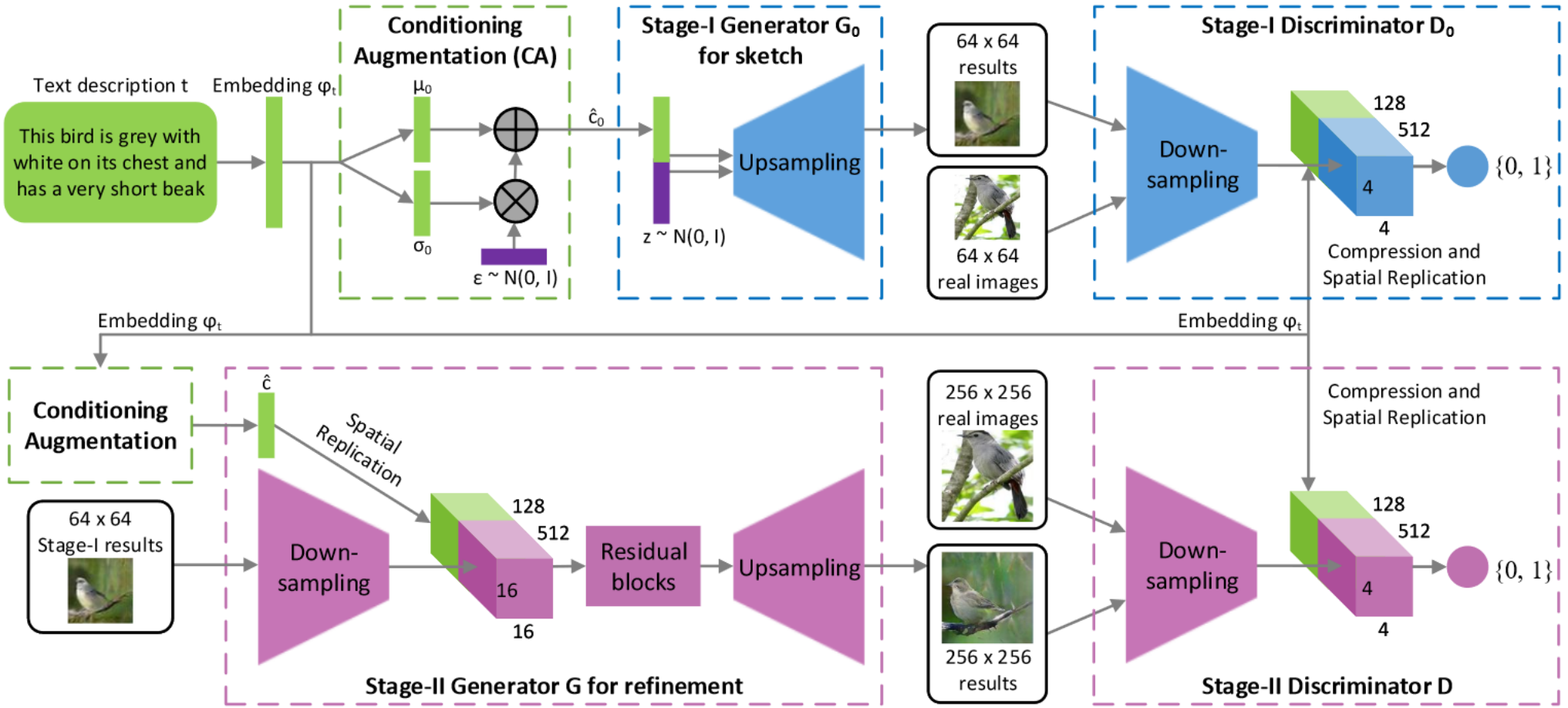
[Karras et al. CVPR ‘19 - A Style-Based Generator Architecture for Generative Adversarial Networks]

StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks



[Zhang et al. ICCV '17 - StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks]

Architecture

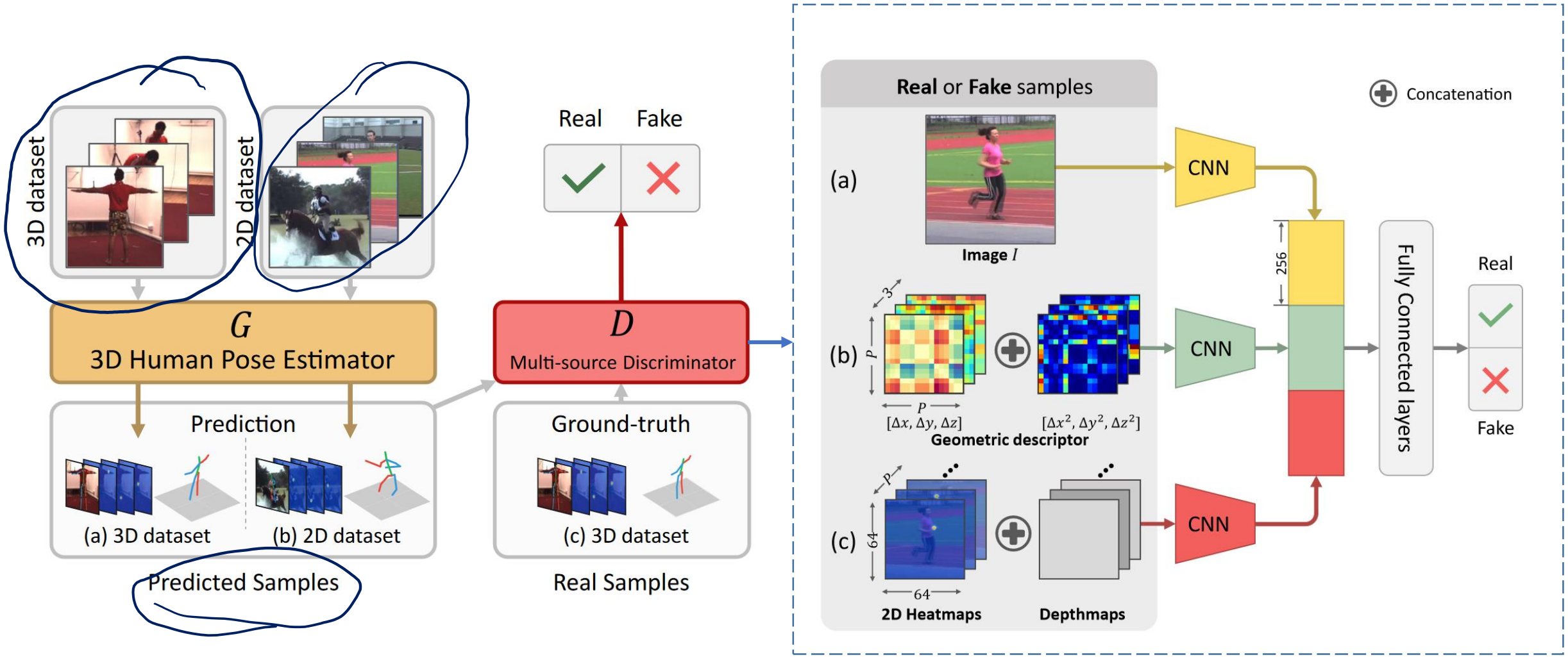


[Zhang et al. ICCV '17 - StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks]

More qualitative examples

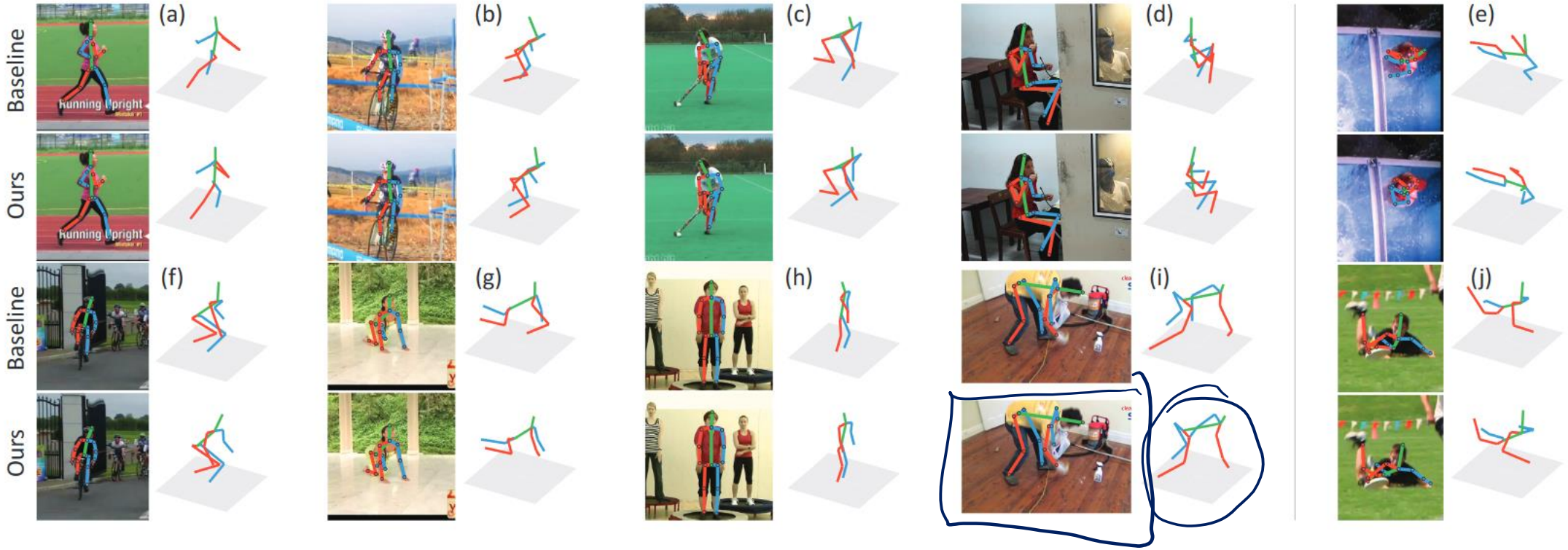
[Zhang et al. ICCV '17 - StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks]

3D Human Pose Estimation in the Wild by Adversarial Learning



[Yang et al. CVPR '18 - 3D Human Pose Estimation in the Wild by Adversarial Learning]

Result of adversarial training



[Yang et al. CVPR '18 - 3D Human Pose Estimation in the Wild by Adversarial Learning]

Summary

GANs have [theoretical] ability to model distributions fully

Often generating consistent samples is enough

State-of-the-art results on many tasks such as image-to-image translation

Higher image quality compared to VAEs

Key aspect is the “perceptual” loss emulated by the discriminator

Next Week

GAN ++

Autoregressive Models