《物理与人工智能》

10. 举例-生成对抗网络

授课教师： 马滟青

2025/10/13（第五周）

鸣谢：基于slazebni幻灯片

北京大學

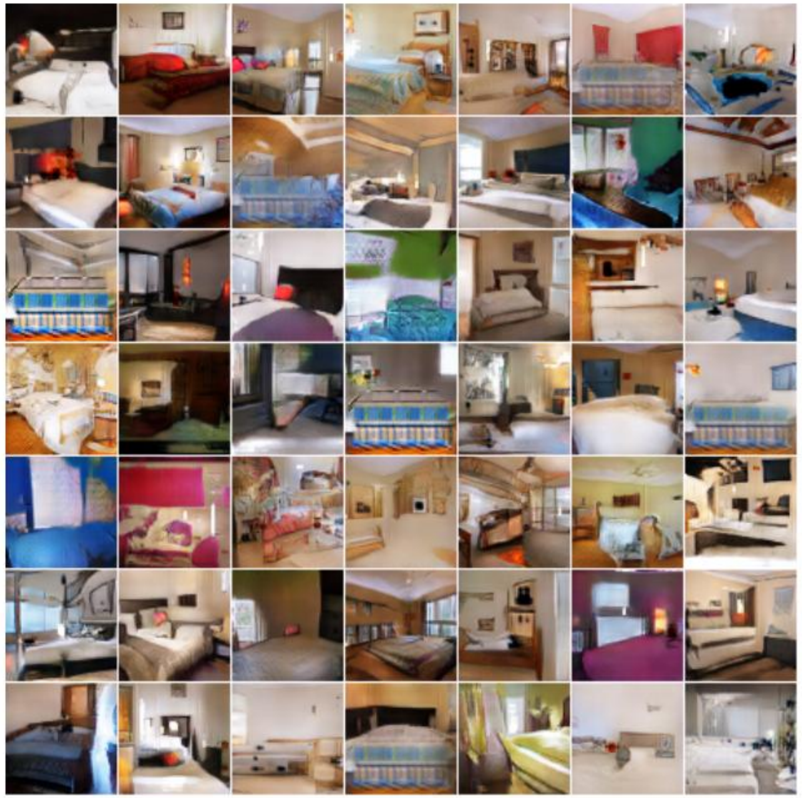

头条 @零凝

Outline

- Generative modeling tasks
- Original GAN formulation
- Alternative GAN objectives

Generative modeling tasks

- Generation: learn to sample from the distribution represented by the training set



Generative modeling tasks

- Generation conditioned on class label or text prompt

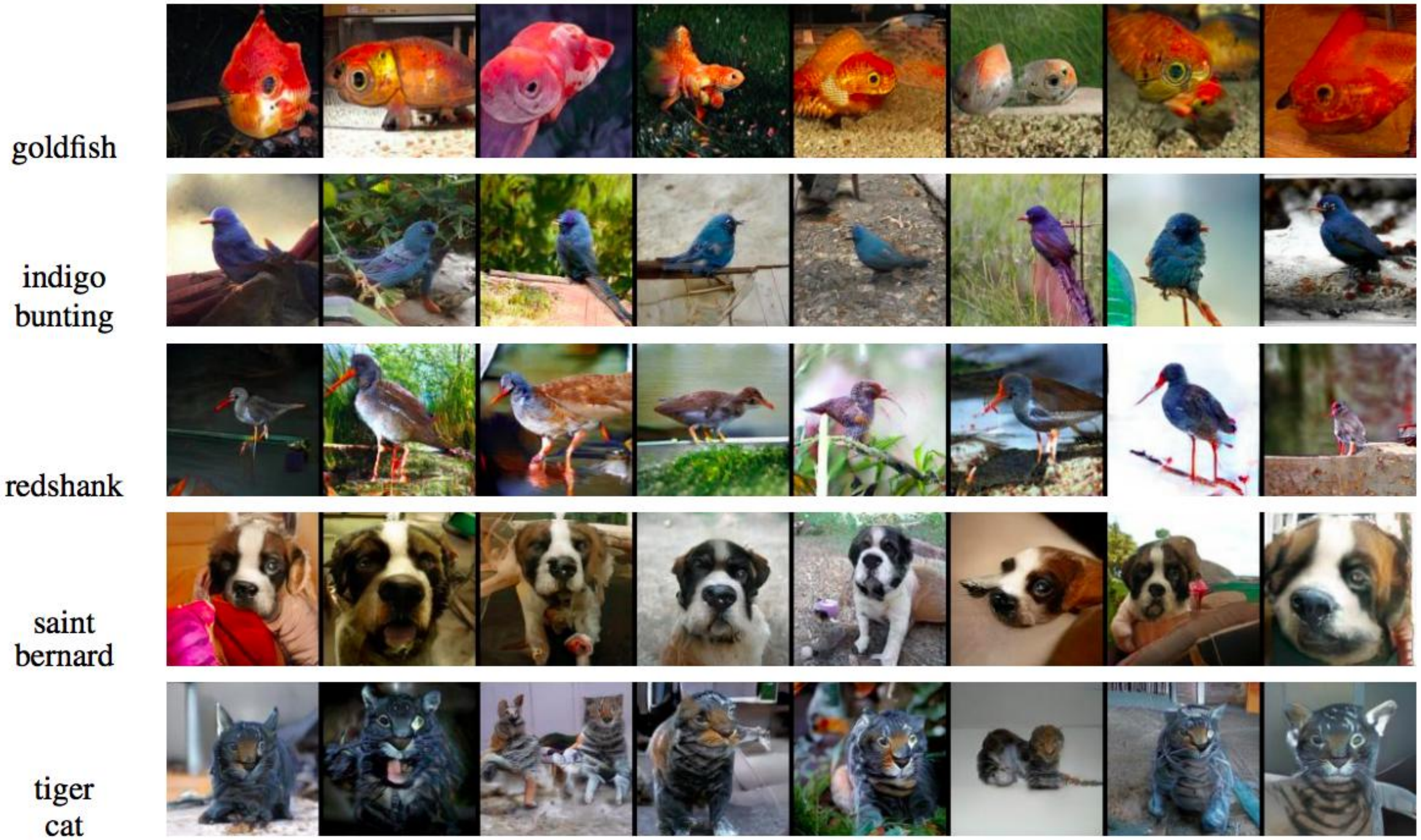
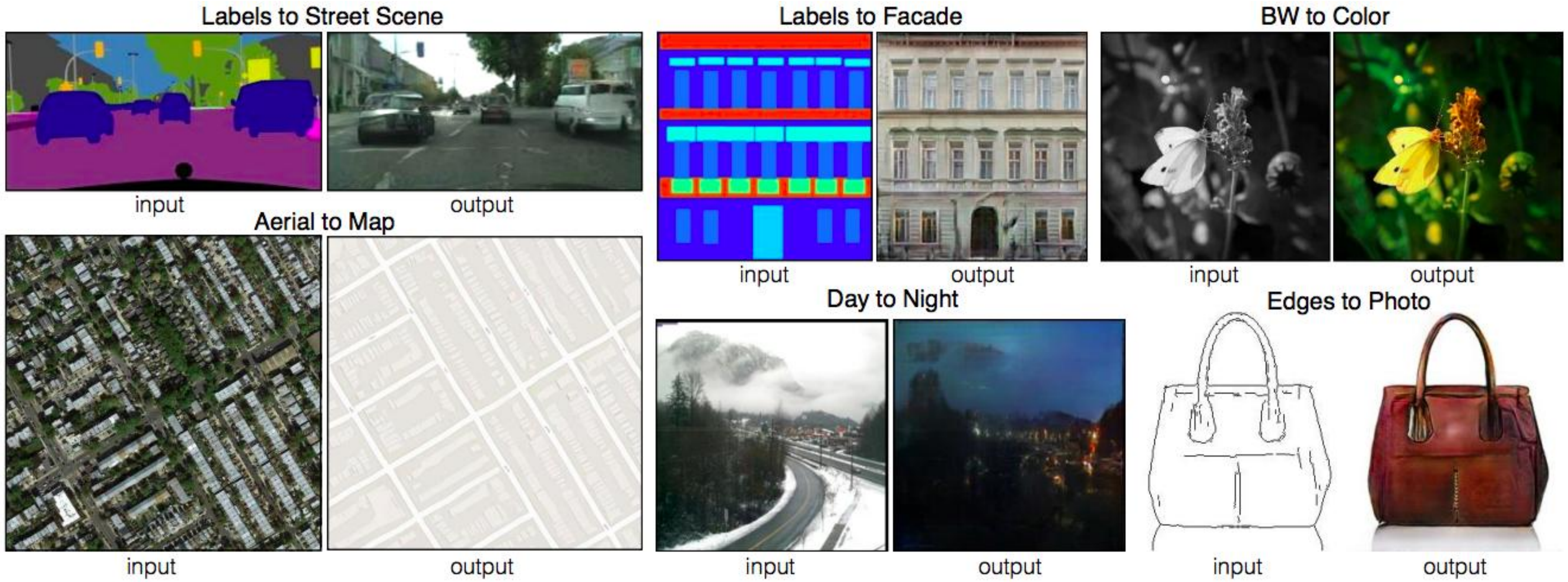


Figure source

Generative modeling tasks

- Generation conditioned on image (*image-to-image translation*)



P. Isola, J.-Y. Zhu, T. Zhou, A. Efros, Image-to-Image Translation with Conditional Adversarial Networks, CVPR 2017

Designing a network for generative tasks

1. We need an architecture that can generate an image

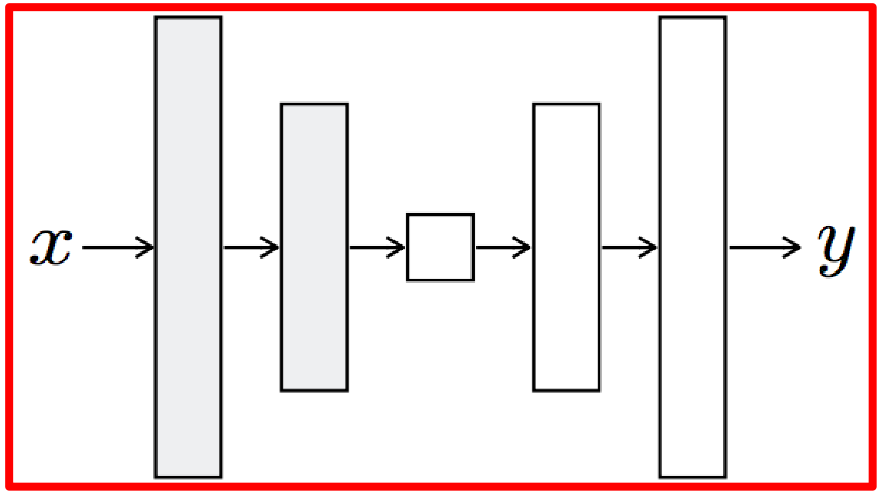
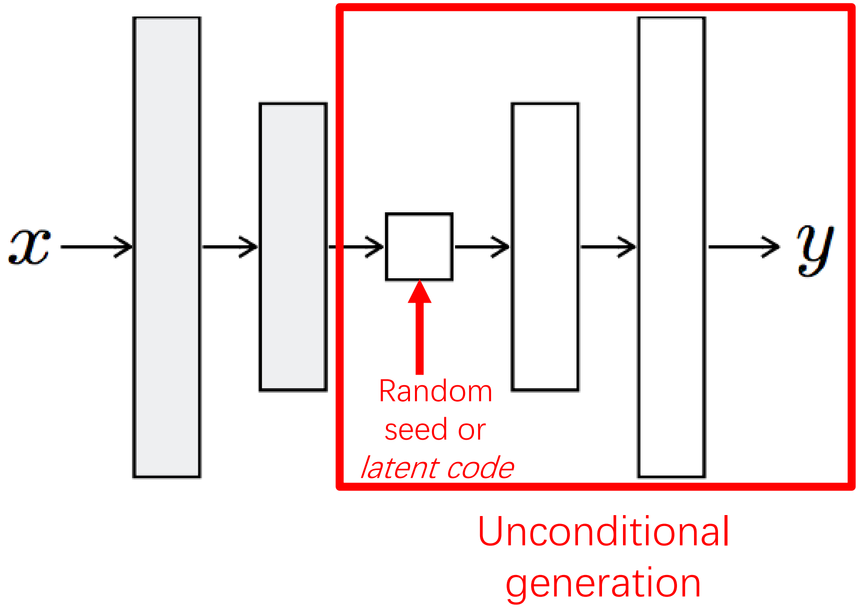


Image-to-image translation

Designing a network for generative tasks

1. We need an architecture that can generate an image



Designing a network for generative tasks

1. We need an architecture that can generate an image
2. We need to design the right loss function and training framework

Learning to sample

Training data $x \sim p_{\text{data}}$

Generated samples $x \sim p_{\text{model}}$

We want to learn p_{model} that matches p_{data}

Adapted from Stanford CS231n

Generative adversarial networks

- Train two networks with opposing objectives:
  - **Generator:** learns to generate samples
  - **Discriminator:** learns to distinguish between generated and real samples

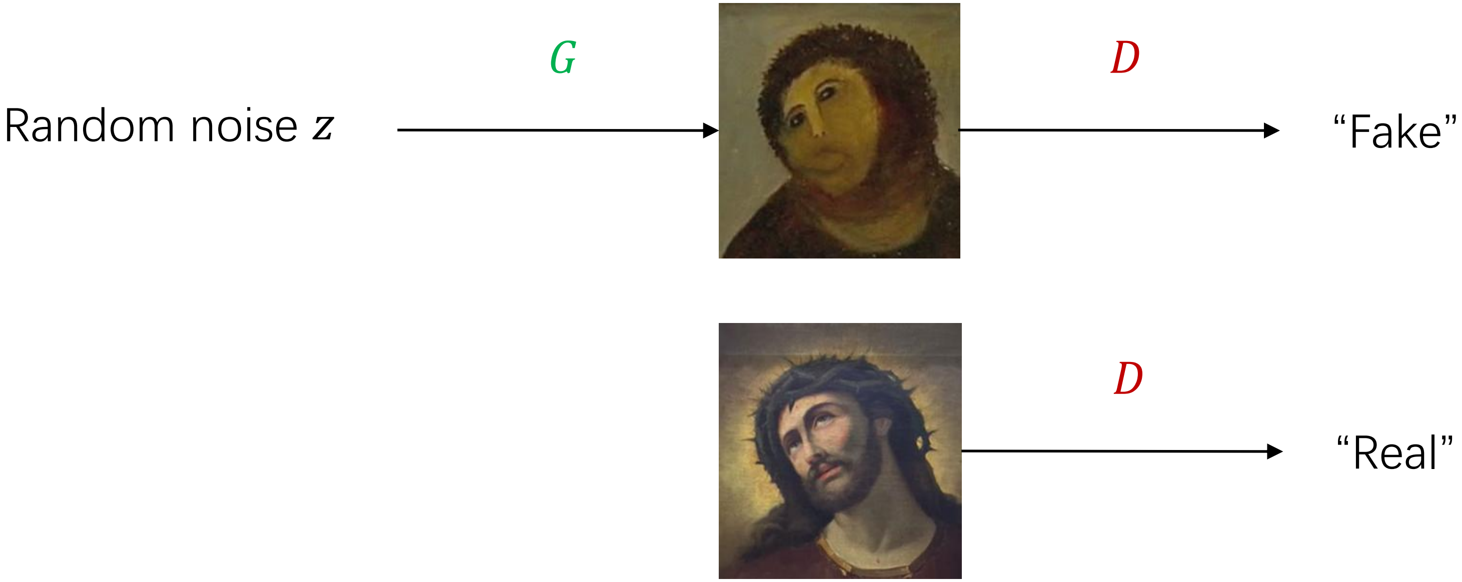
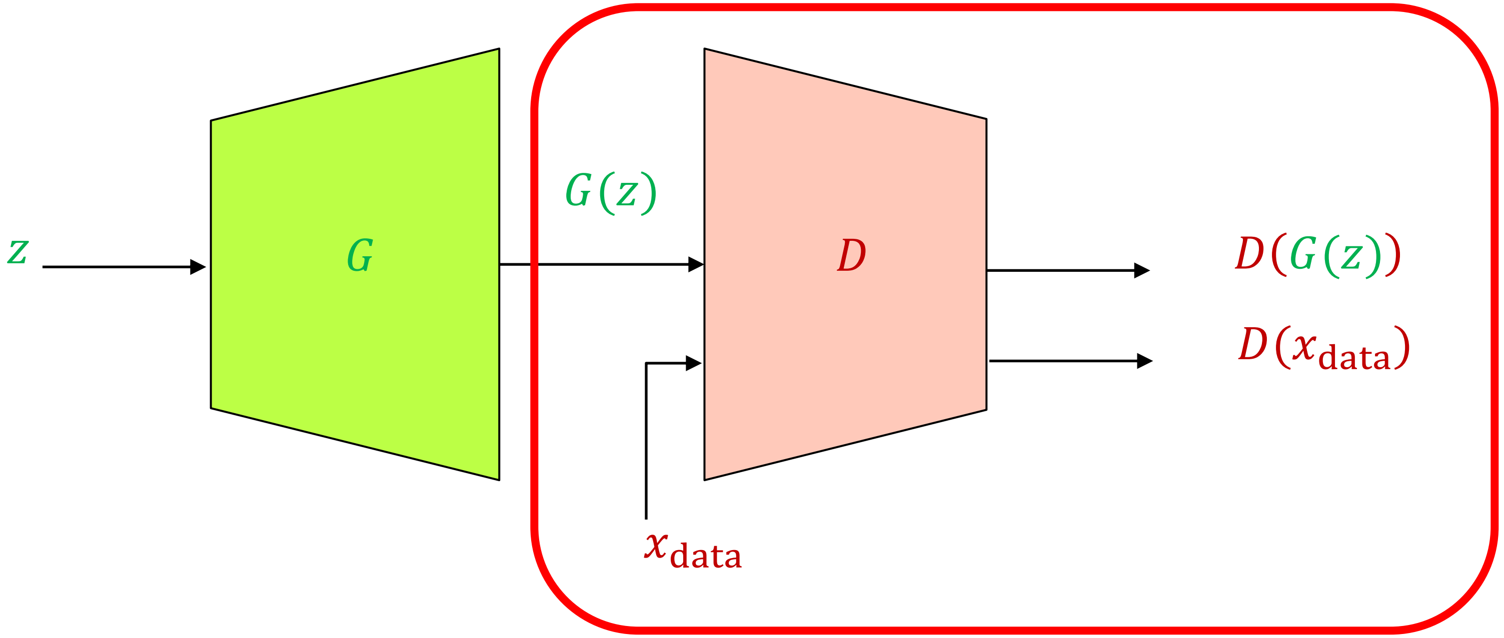


Figure adapted from [F. Fleuret]

I. Goodfellow et al. [Generative adversarial nets]. NeurIPS 2014

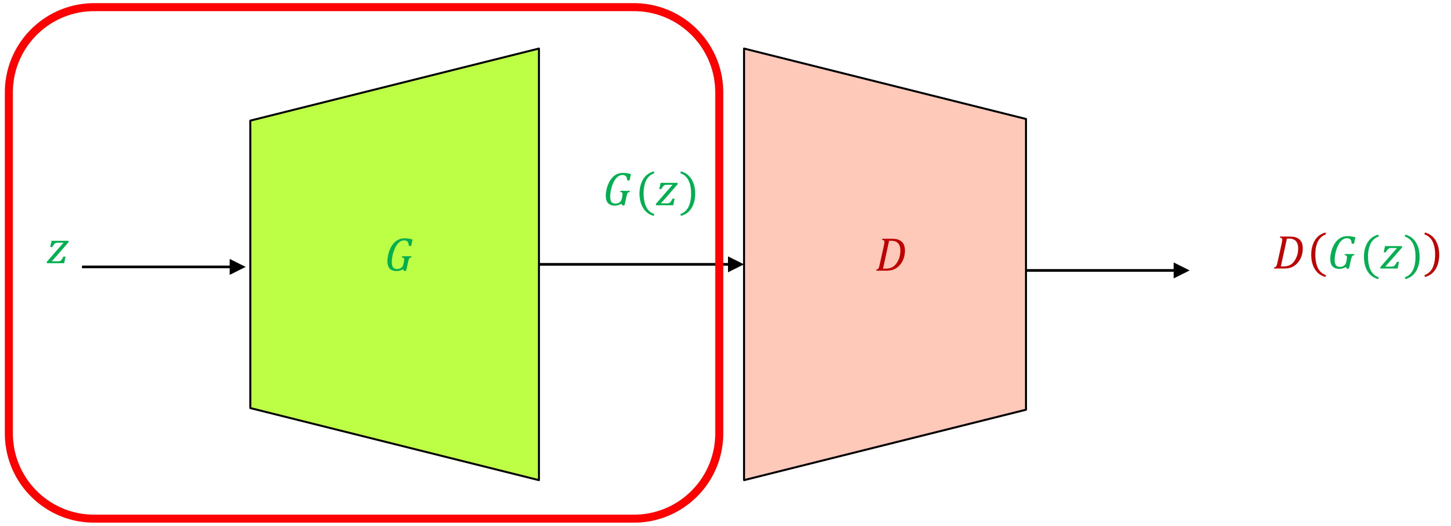
GAN: Schematic picture

- Update discriminator: push $D(x_{\text{data}})$ close to 1 and $D(G(z))$ close to 0
  - The generator is a "black box" to the discriminator



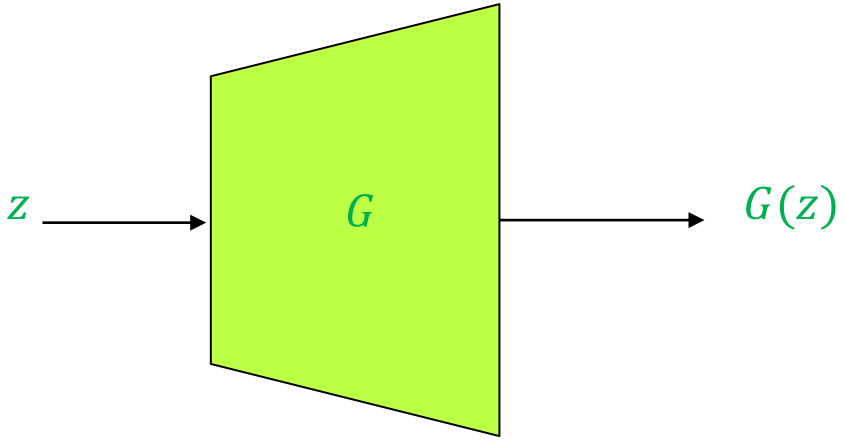
GAN: Schematic picture

- Update generator: increase $D(G(z))$
  - Requires back-propagating through the composed generator-discriminator network (i.e., the discriminator cannot be a black box)
  - The generator is exposed to real data only via the output of the discriminator *and its gradients*



GAN: Schematic picture

- Test time – the discriminator is discarded



Original GAN results

MNIST digits

Toronto Face Dataset

Nearest real image for sample to the left

I. Goodfellow et al. Generative adversarial nets. NeurIPS 2014

Original GAN results

CIFAR-10 (FC networks)

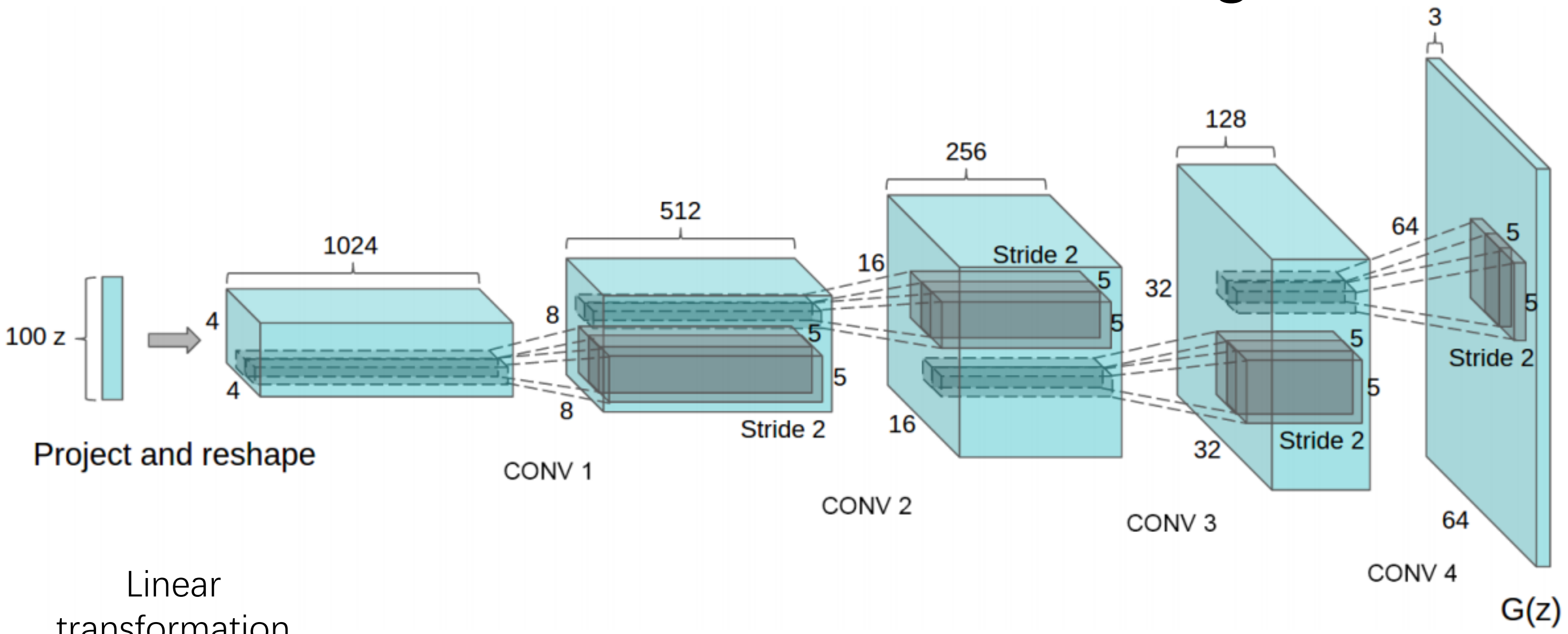
CIFAR-10 (conv networks)

I. Goodfellow et al. Generative adversarial nets. NeurIPS 2014

DCGAN

- Early, influential convolutional architecture for generator

Uniformly distributed input



Linear transformation

Four transposed convolution layers with output stride of 2 for upsampling, followed by ReLU activations

Tanh activations in the last layer

A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, ICLR 2016

DCGAN

北京大学 PEKING UNIVERSITY

- Early, influential convolutional architecture for generator
- Discriminator architecture (empirically determined to give best training stability):
  - Don't use pooling, only strided convolutions
  - Use Leaky ReLU activations (sparse gradients cause problems for training)
  - Use only one FC layer before the softmax output
  - Use batch normalization after most layers (in the generator also)

A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, ICLR 2016

DCGAN results

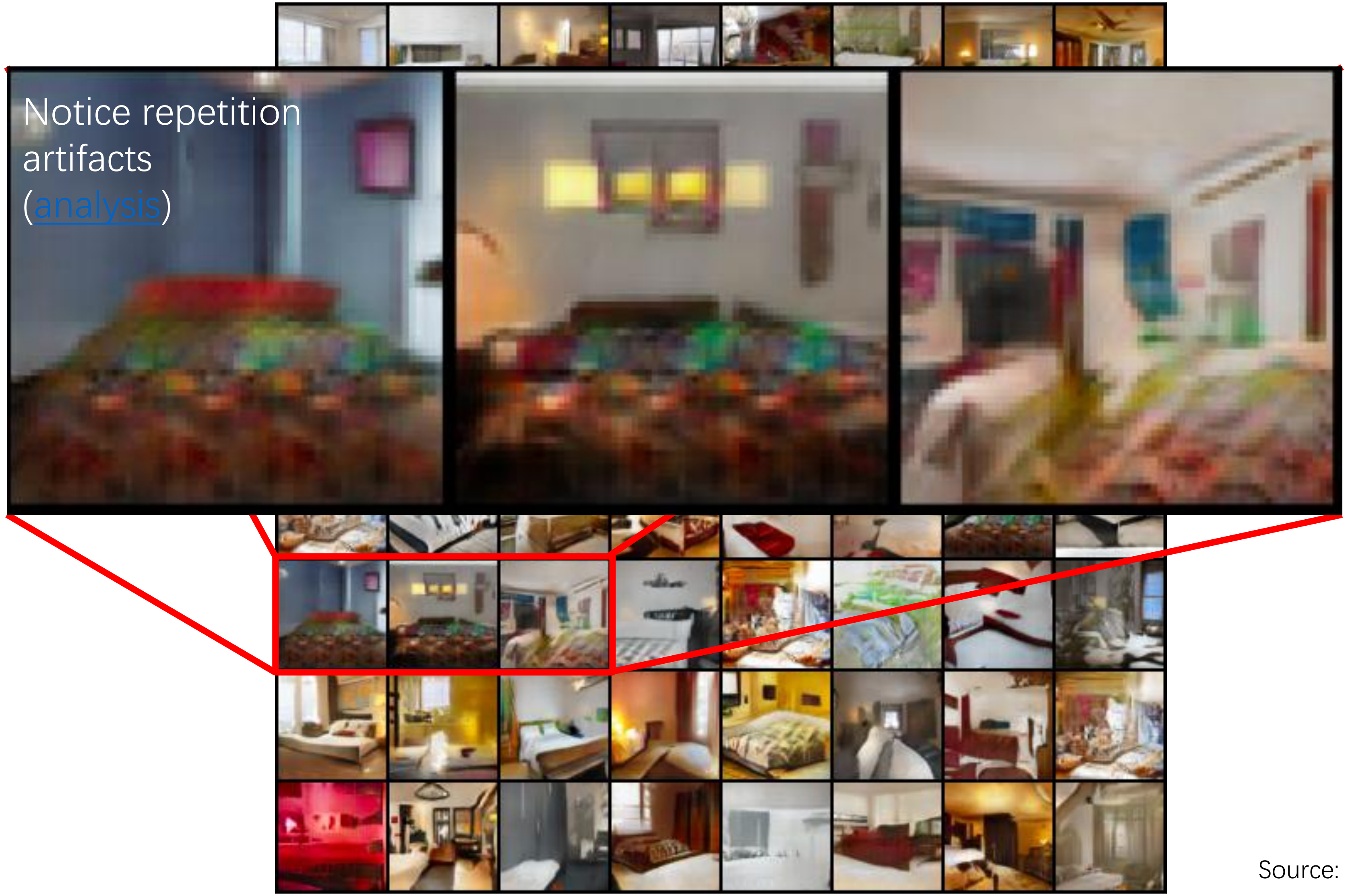
Generated bedrooms after one epoch

DCGAN results

Generated bedrooms after five epochs

DCGAN results

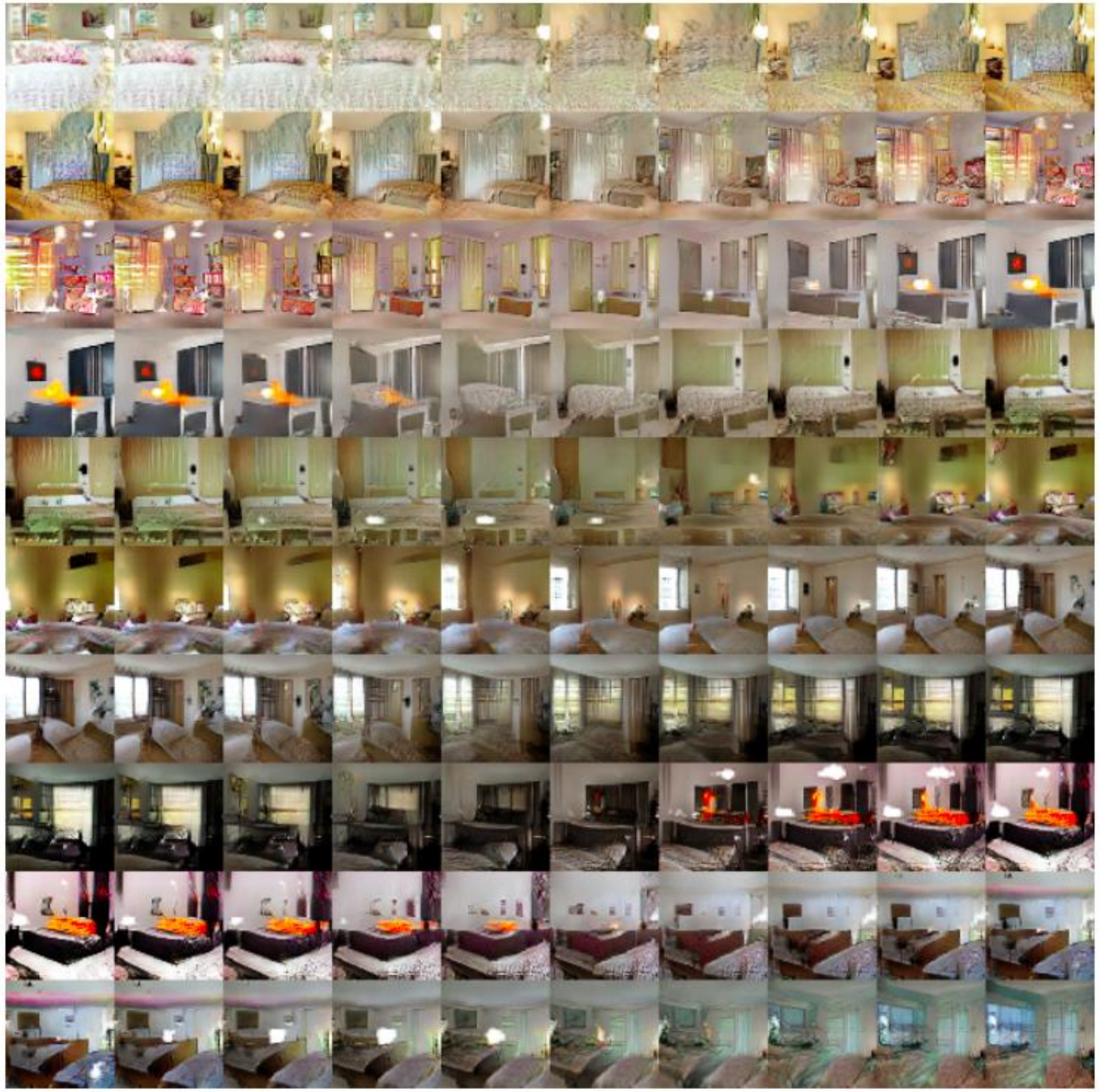
More bedrooms



Source: F. Fleuret

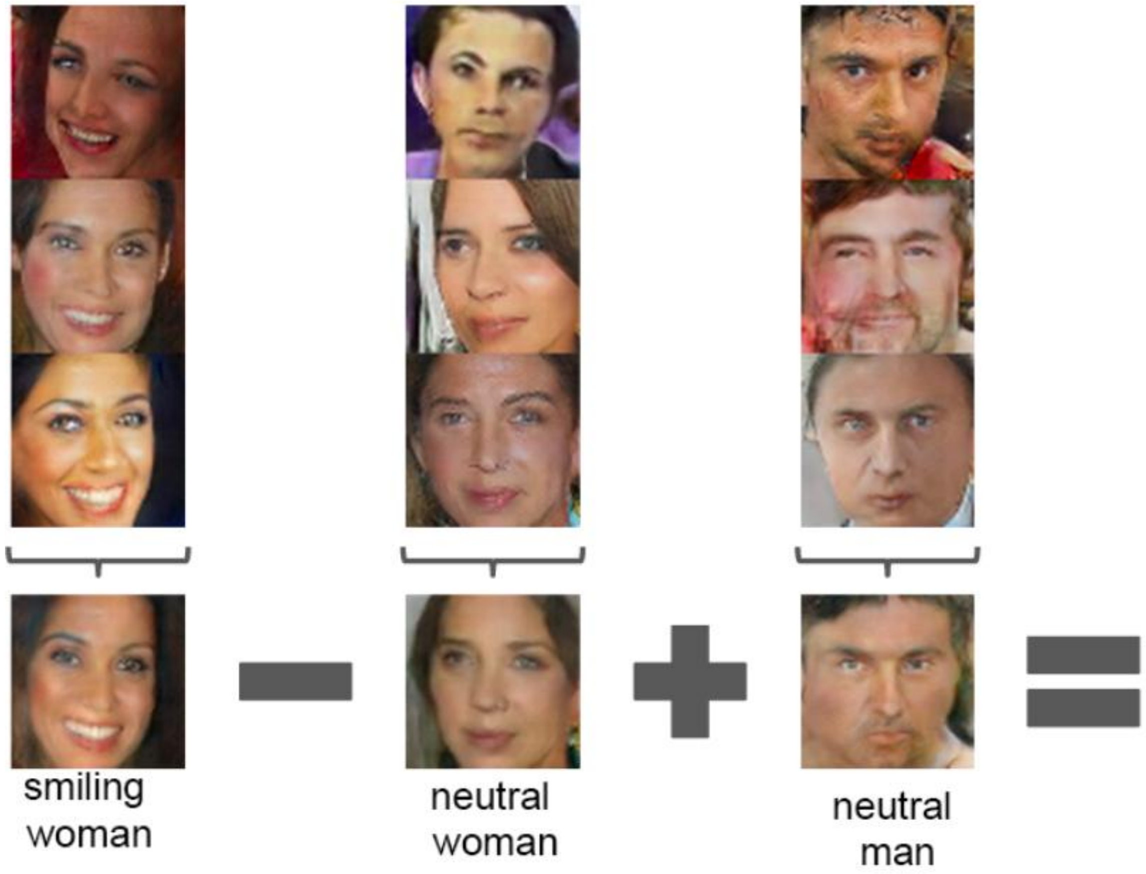
DCGAN results

Interpolation between different points in the z space



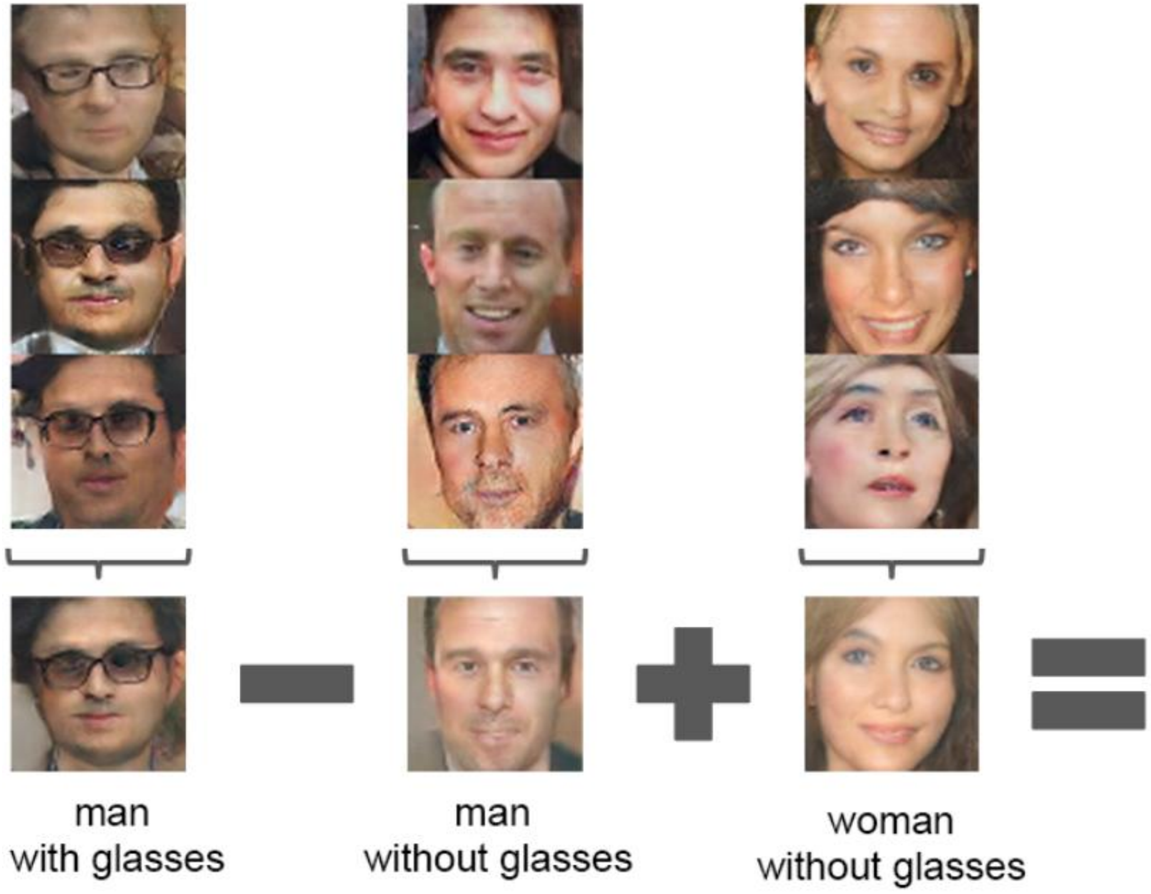
DCGAN results

- Vector arithmetic in the z space



DCGAN results

- Vector arithmetic in the z space



DCGAN results

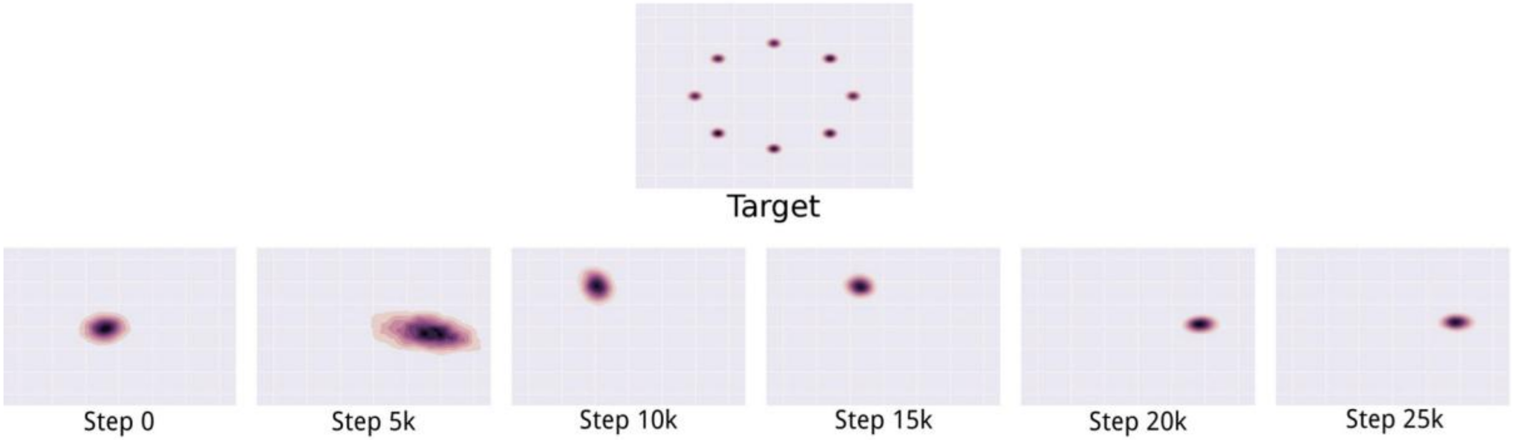
- Pose transformation by adding a “turn” vector

Problems with GAN training

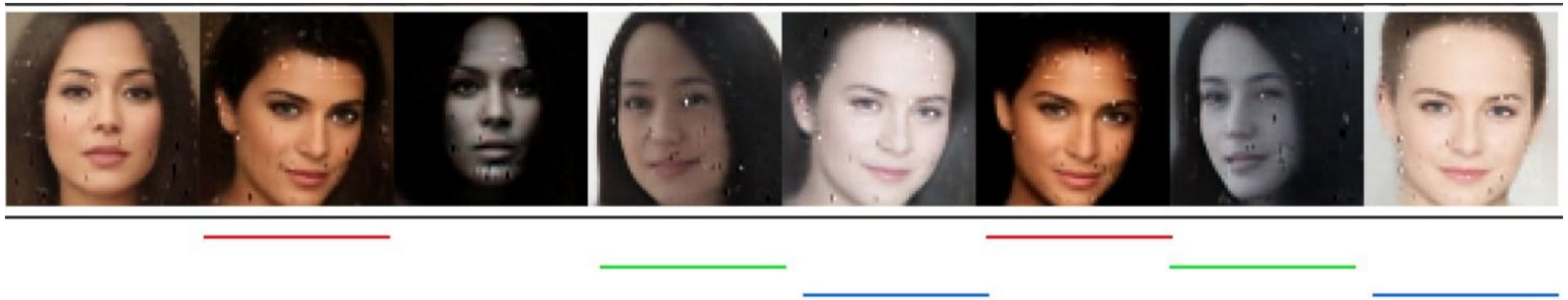
- Stability
  - Parameters can oscillate or diverge, generator loss does not correlate with sample quality
  - Behavior very sensitive to hyperparameter selection

Problems with GAN training

- Mode collapse
  - Generator ends up modeling only a small subset of the training data



Source



Source

Outline

北京大學 PEKING UNIVERSITY

- Generative modeling tasks
- Original GAN formulation
- Alternative GAN objectives

Wasserstein GAN (WGAN)

- Motivated by *Wasserstein or Earth mover's distance*, which is an alternative to JS divergence for comparing distributions
  - In practice, use linear activation instead of sigmoid in the discriminator and drop the logs from the objective:

$$\min_G \max_D \left[\mathbb{E}_{x\sim p_{\text{data}}} D(x) - \mathbb{E}_{z\sim p} D(G(z))\right]$$

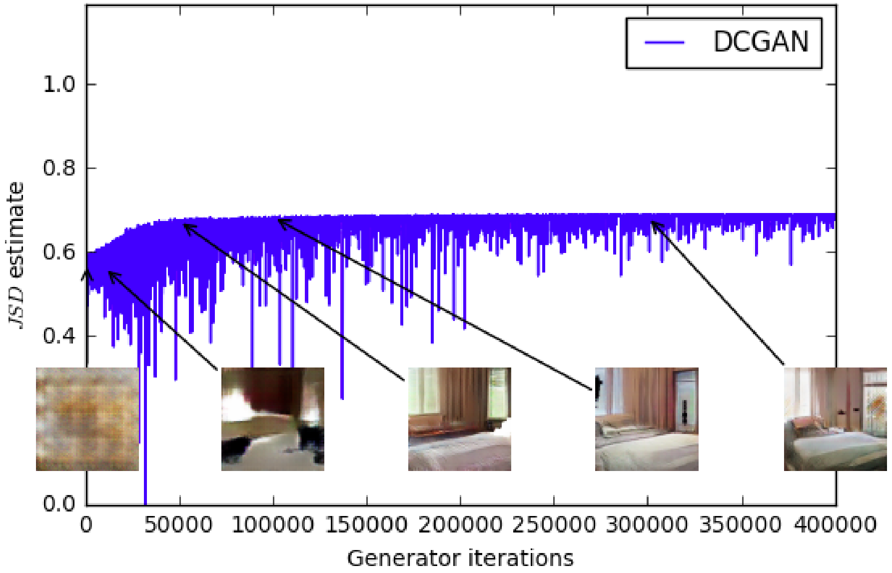
  - Due to theoretical considerations, important to ensure smoothness of discriminator
  - This paper's suggested method is clipping weights to fixed range $[-c, c]$

M. Arjovsky, S. Chintala, L. Bottou, Wasserstein generative adversarial networks, ICML 2017

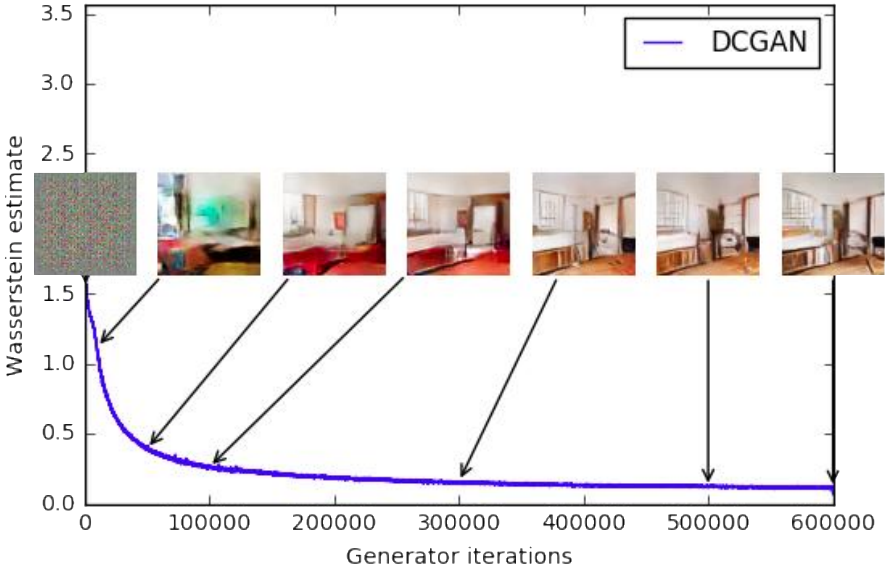
Wasserstein GAN (WGAN)

- Benefits (claimed)
  - Better gradients, more stable training
  - Objective function value is more meaningfully related to quality of generator output

Original GAN divergence



WGAN divergence



M. Arjovsky, S. Chintala, L. Bottou, Wasserstein generative adversarial networks, ICML 2017

Improved Wasserstein GAN (WGAN-GP)

北京大学 PEKING UNIVERSITY

- Weight clipping leads to problems with discriminator training
- Improved Wasserstein discriminator loss:

$$\mathbb{E}_{\tilde{x}\sim p_{\text{gen}}} D(\tilde{x}) - \mathbb{E}_{x\sim p_{\text{real}}} D(x)$$

$$+\lambda\, \mathbb{E}_{\hat{x}\sim p_{\hat{x}}}\left[\left(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1\right)^2\right]$$

Unit norm gradient penalty on points $\hat{x}$ obtained by interpolating real and generated samples

I. Gulrajani et al. Improved training of Wasserstein GANs. NeurIPS 2017

Improved Wasserstein GAN: Results

I. Gulrajani et al. Improved training of Wasserstein GANs. NeurIPS 2017

Least Squares GAN (LSGAN)

- Use least squares cost for generator and discriminator
  - Equivalent to minimizing Pearson χ^2 divergence

$$L_D = \mathbb{E}_{x\sim p_{\text{data}}}(D(x)-1)^2 + \mathbb{E}_{z\sim p}(D(G(z)))^2$$

Push discrim. response on real data close to 1

Push response on generated data close to 0

$$L_G = \mathbb{E}_{z\sim p}(D\big(G(z)\big)-1)^2$$

Push response on generated data close to 1

X. Mao et al. Least squares generative adversarial networks. ICCV 2017

Least Squares GAN (LSGAN)

- Benefits (claimed)
  - Higher-quality images

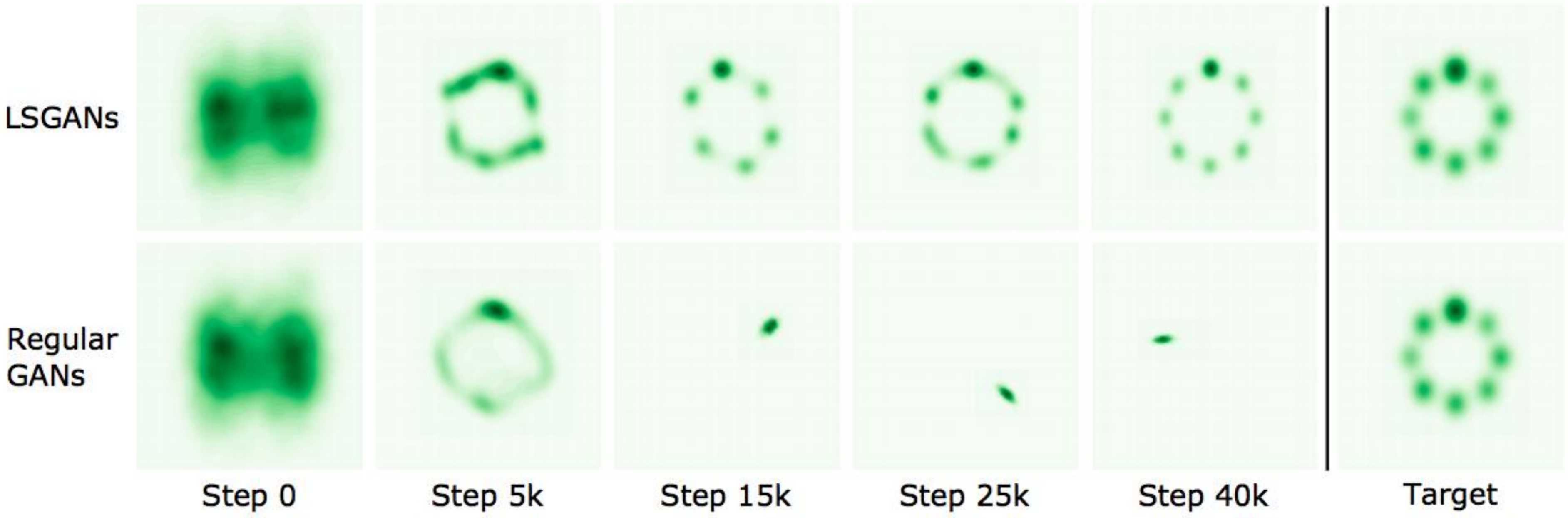
(a) Generated images (112 × 112) by LSGANs.

(b) Generated images (112 × 112) by DCGANs.

X. Mao et al. [Least squares generative adversarial networks](). ICCV 2017

Least Squares GAN (LSGAN)

- Benefits (claimed)
  - Higher-quality images
  - More stable and resistant to mode collapse



X. Mao et al. [Least squares generative adversarial networks](). ICCV 2017

GAN with hinge loss

- Discriminator: Drive discriminator score on real data above 1, on generated data below -1

$$L_D = -\mathbb{E}_{x\sim p_{\text{data}}}[\min(0, D(x) - 1)] \\ -\mathbb{E}_{z\sim p}\left[\min(0, -D\left(G(z)\right) - 1)\right]$$

- Generator: maximize discriminator score on generated data

$$L_G = -\mathbb{E}_{z\sim p} D(G(z))$$

T. Miyato et al. Spectral normalization for generative adversarial networks. ICLR 2018

谢
谢

北京大学
PEKING UNIVERSITY