Generative Adversarial Networks

Speaker: Mingde Wang

Tutor: Professor Qiao Yan

Outline

- Why study **generative models**?
- The principle of GAN.
- GAN application.

Creation

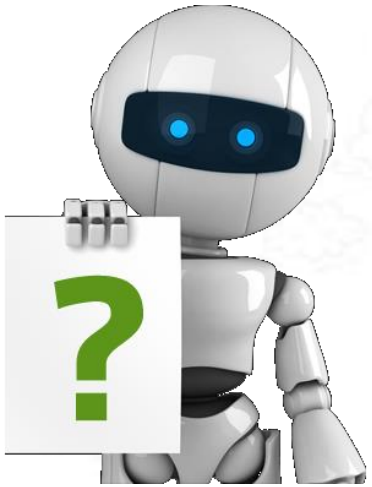
One of our core aspirations at AI is to develop algorithms and techniques that endow computers with an understanding of our world.

"What I cannot create, I do not understand."

—Richard Feynman

If you don't know how to produce a thing, you don't really understand it.

Creation – Image Processing



Classification

V.S

More applications

Machine draws a cat

Generative models

Generative models, a branch of unsupervised learning techniques in machine learning. Unlike traditional classification models. Generative models are one of the most promising approaches towards this goal that develop models and algorithms that can analyze and understand these real-world data.

1. Excellent test of our ability to use high-dimensional,complicated probability distributions.
2. Simulate possible futures for planning or simulated RL.
3. Missing data - semi-supervised learning.
4. Multi-modal outputs.
5. Realistic generation tasks.

Generative models

PixelRNN – a simple example

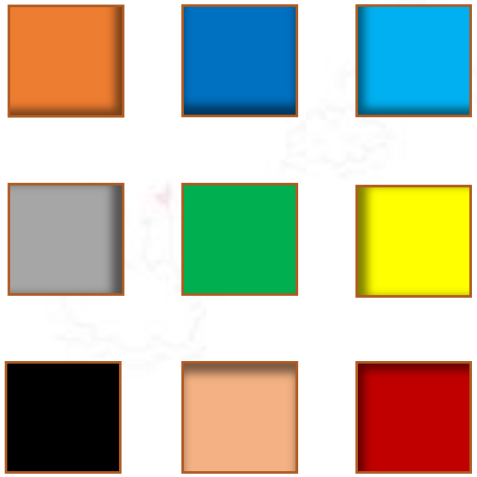
Generative Adversarial Network(GAN)

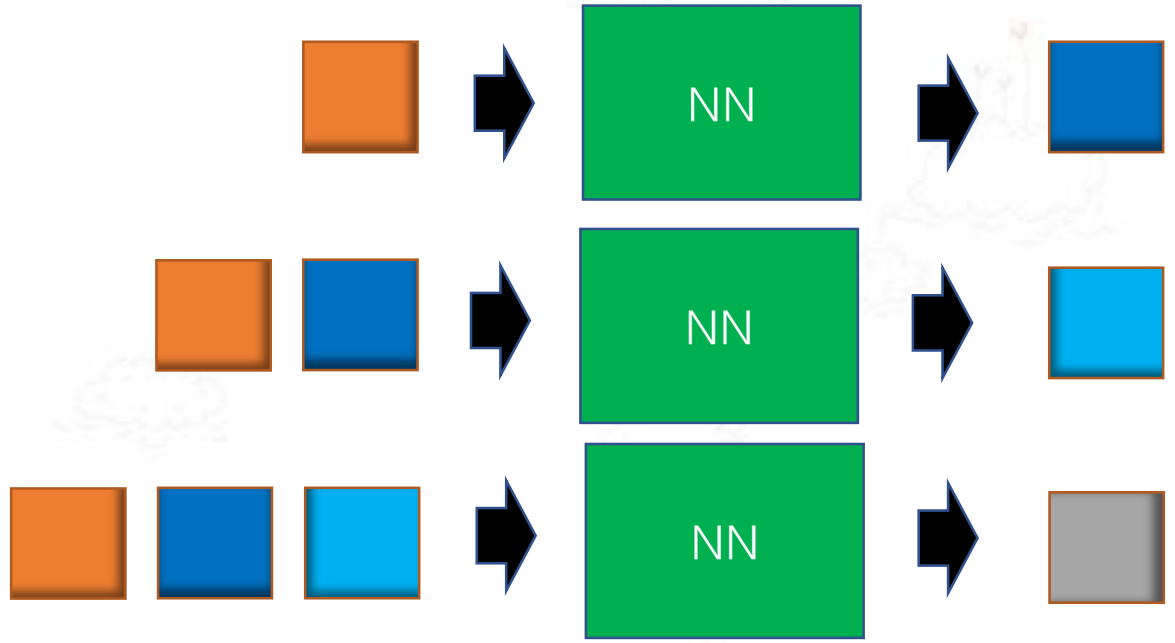
PixelRNN

Ref: Aaron van den Oord, Nal Kalchbrenner, Koray Kavukcuoglu, Pixel Recurrent Neural Networks, arXiv preprint, 2016

To create an image, generating a pixel each time

E.g. 3 X 3 image

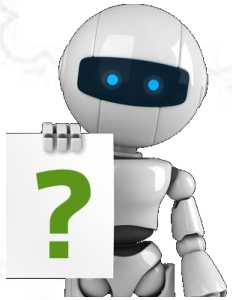


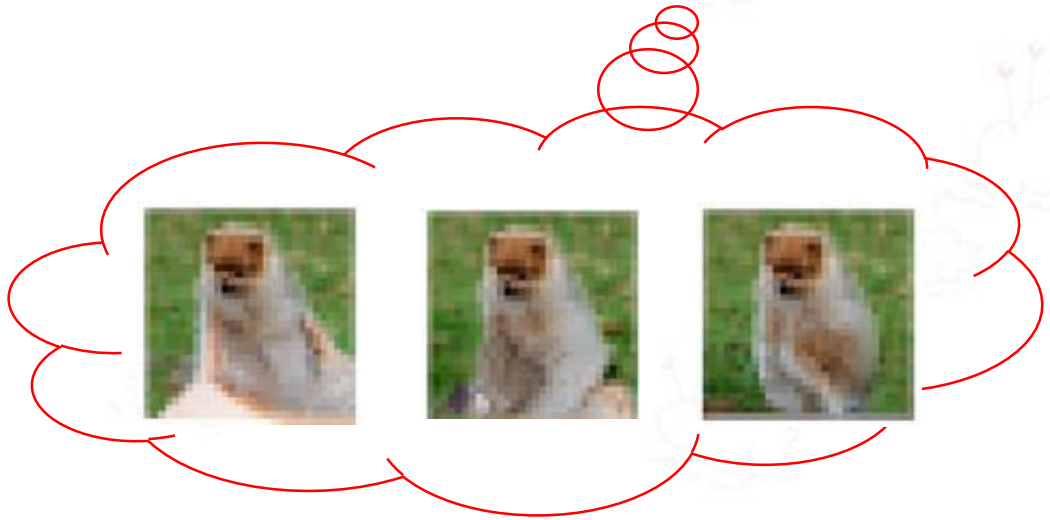


Can be trained just with a large collection of images without any annotation.

PixelRNN

Real world





Problem of PixelRNN

- sample quality is not high.
- It is difficult to define an effective probability density.
- No paper has successfully used PixelRNN for unsupervised/semi-supervised feature learning.

Generative Adversarial Network

Butterfly is not brown

Butterfly has no veins

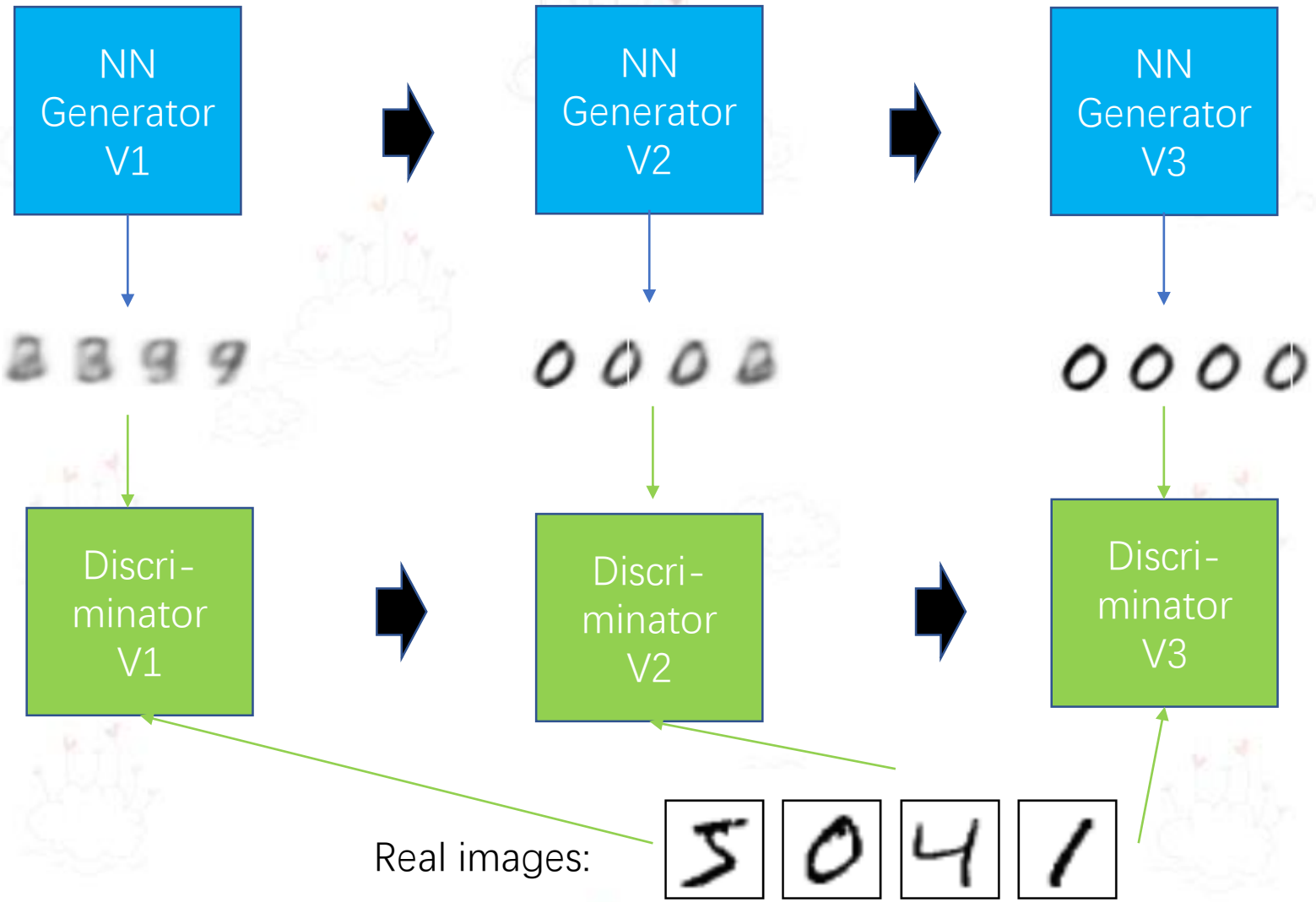
……

Generative Adversarial Network

Generative Adversarial Networks (GANs) are a powerful class of generative models that cast generative modeling as a game between two networks: a generator (e.g. Butterfly) network produces synthetic data given some noise source and a discriminator (e.g. Natural enemy) network discriminates between the generator’s output and true data.

MNIST

The evolution of generation

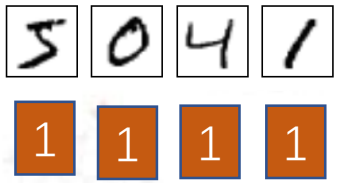


GAN - Discriminator

Vectors z form a distribution ($z \sim N(\mu, \sigma)$)

NN Generator V1

Real images:



image

Discri-minator V1

1/0 (real or fake)

GAN - Generator

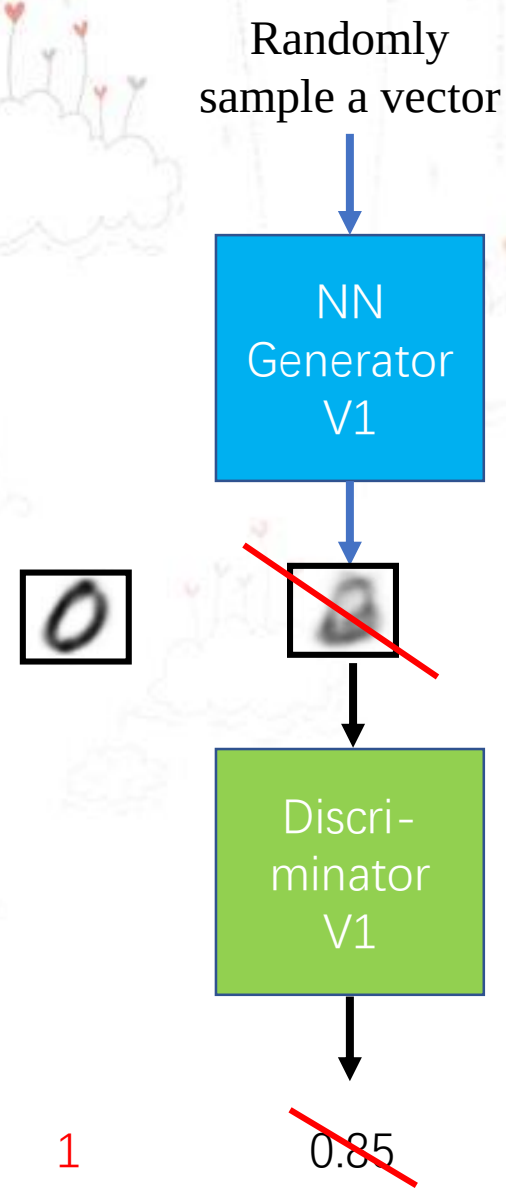
"Tuning" the parameters of generator

➡ The output be classified as "real" (as close to 1 as possible)

Generator + Discriminator = a network

Using gradient descent to find the parameters of generator

Fix the discriminator



Application

1. Images processing
   - Generation High-Quality images
   - Image inpainting
   - Super-Resolution
   - Image translation
   - Facial attribute manipulation
   - …
2. Video prediction and generation
3. Music
4. Natural language processing
   - Semantic Segmentation
5. Improving classification and Recong
   - Learning from Simulated and Unsupervised Images
6. Semi-supervised learning
   - Unsupervised and Semi-supervised Learning
7. Reinforcement learning
8. 3D
9. Medicine
   - Unsupervised Anomaly Detection to Guide Marker Discovery
10. Information security
11. Chess competition
12. …

Ref: https://github.com/zhangqianhui/AdversarialNetsPapers

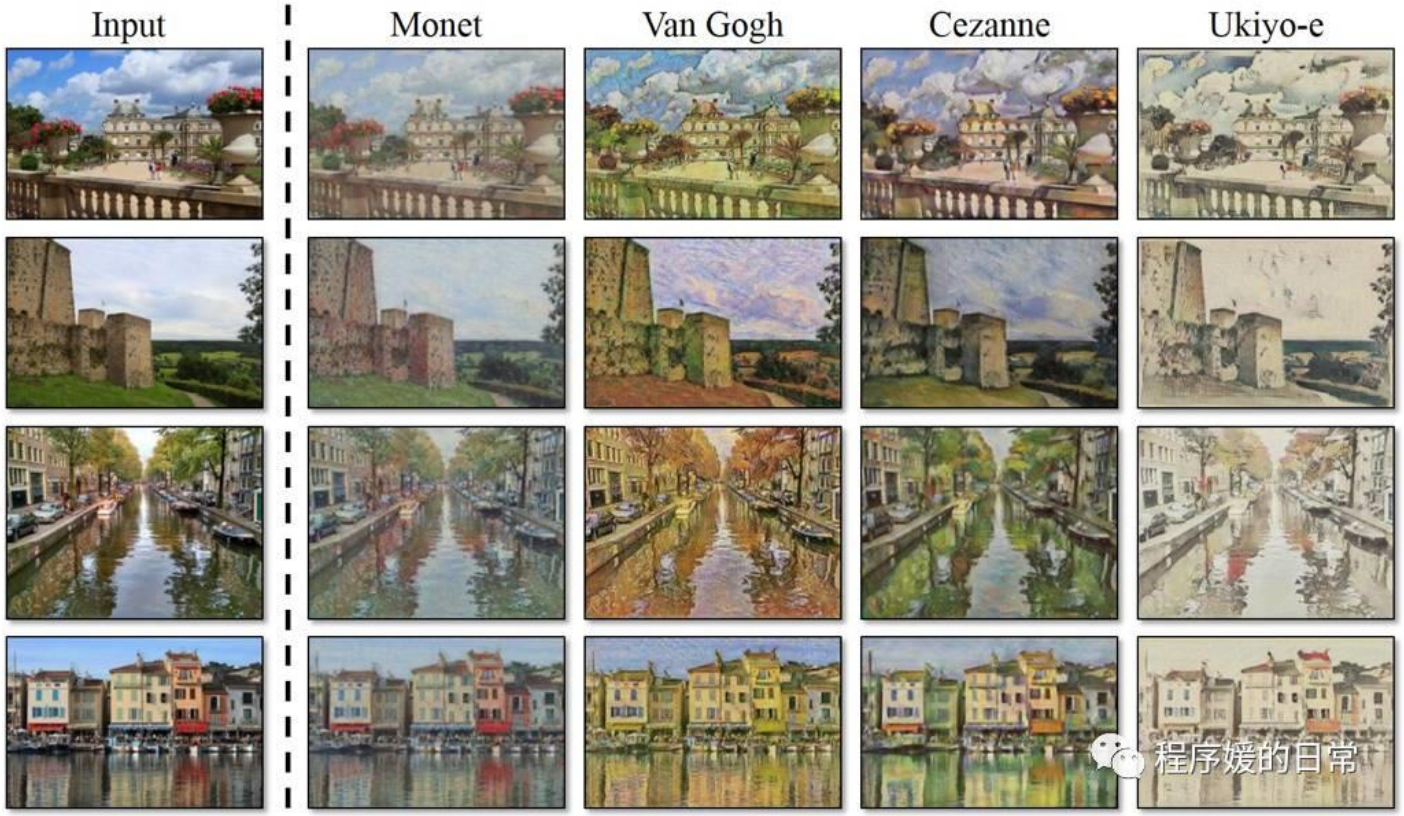
Application

Image inpainting

Ref: http://bamos.github.io/2016/08/09/deep-completion

Application

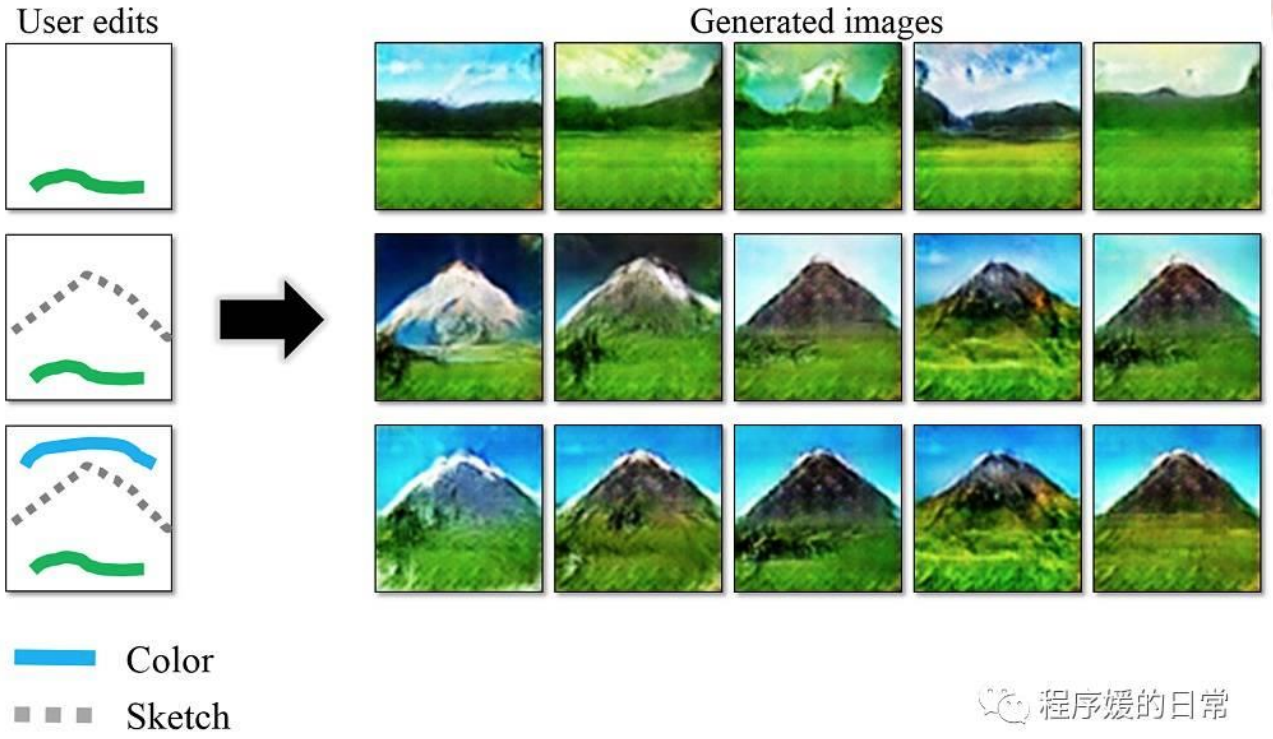
Style conversion based on painter style



Ref: Jun-Yan Zhu*, Taesung Park*, Phillip Isola, Alexei A. Efros. “Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks”. arXiv preprint 2017.

Application

Stick figure



Ref: Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman and Alexei A. Efros. "Generative Visual Manipulation on the Natural Image Manifold", in European Conference on Computer Vision (ECCV). 2016.

END

Thank you !