Imperial College London

Introduction to Generative Adversarial Networks and Applications

Lukas Mosser, Olivier Dubrule

Do you know any of these people?

Do you know any of these people?

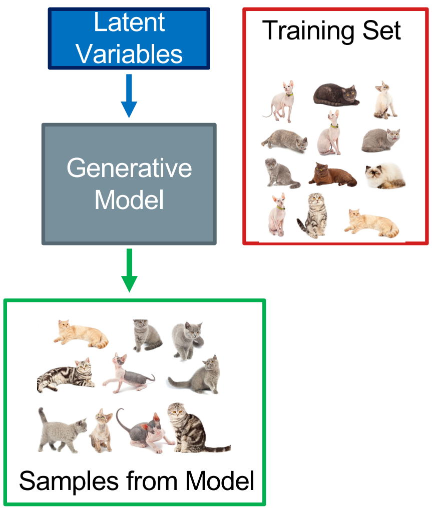
https://thispersondoesnotexist.com/

(Deep) Generative Methods

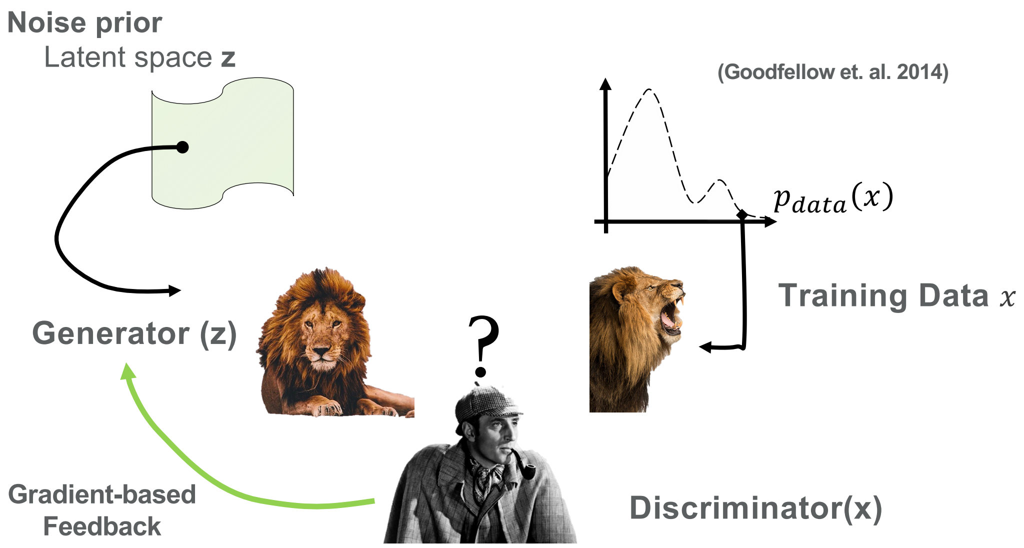
- Task: Draw (new) samples from unknown density given a set of samples

Main Problem: How to find the generative model?

- Generative Adversarial Networks (GAN)
  - Two competing Neural Networks
- Variational Autoencoders (VAE)
  - Bayesian Graphical Model of data distribution
- Autoregression (Pixel-CNN)
  - Conditional Distribution on every sample
- Many More ...



Generative Adversarial Networks – Toy Example



Generative Adversarial Networks – Training

- **Requirements:**
  - **Training Set of data**
  - **Generator – creates samples G(z)** $\mathbf{z} \sim \mathcal{N}(0,1)^{d\times1\times1\times1}$ $G_\theta : \mathbf{z} \rightarrow \mathbb{R}^{1\times64\times64\times64}$
  - **Discriminator – evaluates samples** $D_\omega : \mathbb{R}^{1\times64\times64\times64} \rightarrow [0,1]$
  - **Cost function:** $\min_\theta \max_\omega \{\mathbb{E}_{\mathbf{x}\sim p_{data}}[log\ D_\omega(\mathbf{x})] + \mathbb{E}_{\mathbf{x}\sim p_{\mathbf{z}}}[log(1 - D_\omega(G_\theta(\mathbf{z})))]\}$
- **GAN training – two step procedure in supervised way**
  - **Discriminator training step – Generator fixed**
    - **Train on real data samples**
    - **Train on fake samples**
  - **Generator training step – Discriminator fixed**
    - **Push generator towards "real" images**

Generative Adversarial Networks – Objective

- GAN Objective (Goodfellow 2014): $\min_{\theta} \max_{\omega} \{\mathbb{E}_{\mathbf{x}\sim p_{data}}[log\ D_{\omega}(\mathbf{x})] + \mathbb{E}_{\mathbf{x}\sim p_{\mathbf{z}}}[log(1 - D_{\omega}(G_{\theta}(\mathbf{z})))]\}$

- **For Generator minimize:** $\{\mathbb{E}_{\mathbf{x}\sim p_{data}}[\cancel{log\ D_{\omega}(\mathbf{x})}] + \mathbb{E}_{\mathbf{x}\sim p_{\mathbf{z}}}[log(1 - D_{\omega}(G_{\theta}(\mathbf{z})))]\}$

  -> Minimize: $\mathbb{E}_{\mathbf{x}\sim p_{\mathbf{z}}}[log(1 - D_{\omega}(G_{\theta}(\mathbf{z})))]\}$
  By making $D_{\omega}(G_{\theta}(\mathbf{z}))$ go towards 1!

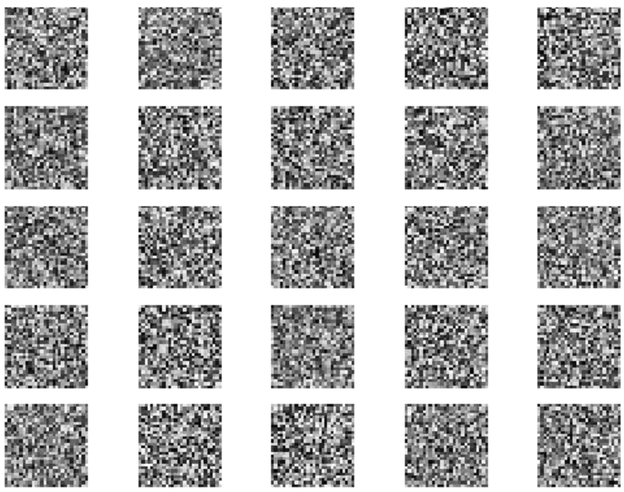
- **Discriminator maximize:** $\{\mathbb{E}_{\mathbf{x}\sim p_{data}}[log\ D_{\omega}(\mathbf{x})] + \mathbb{E}_{\mathbf{x}\sim p_{\mathbf{z}}}[log(1 - D_{\omega}(G_{\theta}(\mathbf{z})))]\}$
  -> Equivalent to the binary cross-entropy

Other Objective functions and even formulations possible e.g. WGAN

GAN Training Example - MNIST

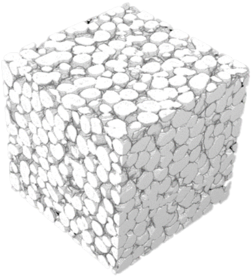
Training Images

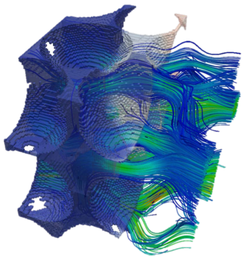
Generative Model (GAN)



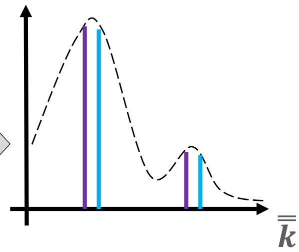
Credit: @eriklindernoren

GANs in practice – Digital Rock and Core Physics





Point Estimates vs Distributions



Data Acquisition
Experiments / Imaging

Flow Experiment
Numerical Simulation

Validation

Challenge:
How to quantify variability?

Perform measurements

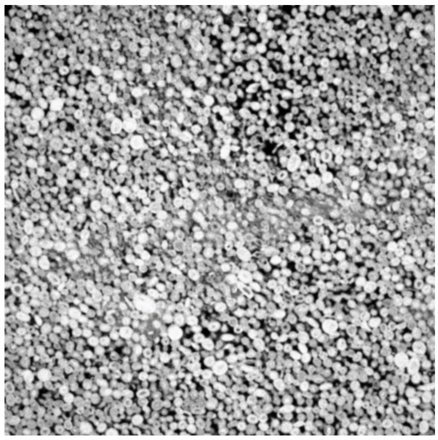
Generate
Representative Images

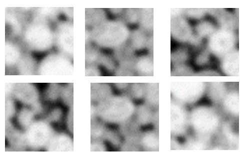
5

Ketton Limestone Dataset and Preprocessing

- Oolitic Limestone
- Intergranular pores
- Intragranular Micro-Porosity
- Ellipsoidal grains
- 99% Calcite
- Image Size:
  - 900^3 voxels @ 26.7 μm

Extract Non-Overlapping
Training Images (64^3 $voxels$)

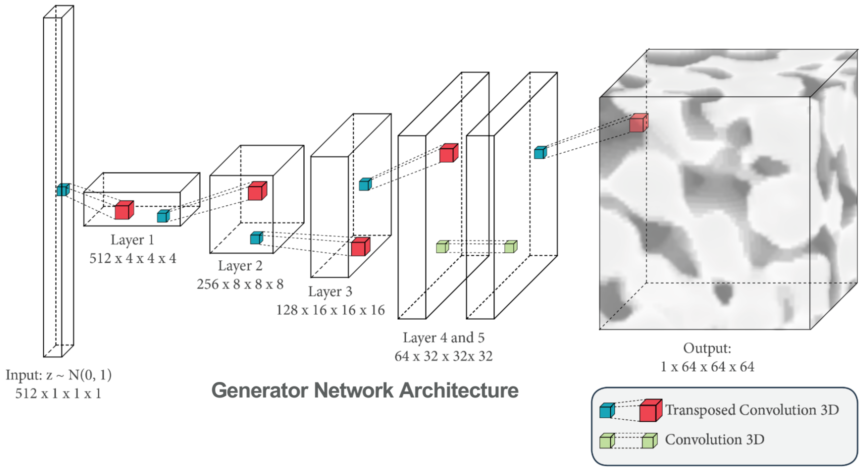




Training Set

Network Architecture - 3D Convolutional Network

Represent G(z) and D(x) as deep neural networks:

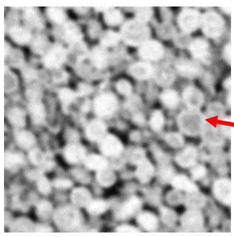


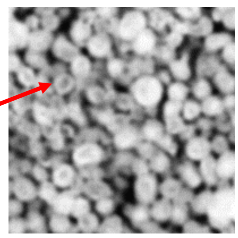
Generator Network Architecture

Discriminator: Binary Classification Network -> Real / Fake

Reconstruction Quality – Unconditional Simulation

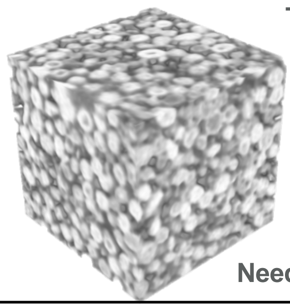
Ketton Training Image | **GAN generated sample**

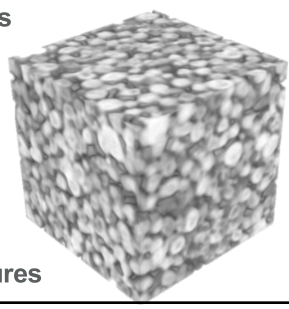




Intergranular Porosity
Moldic Features
Micro-Porosity

Training Time: 8 hours
Generation: 5 sec.

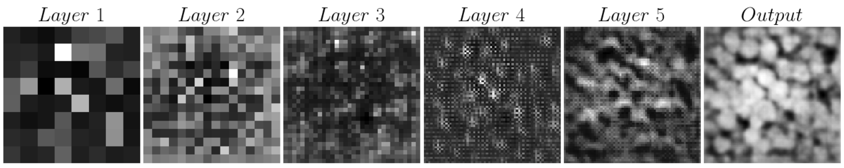




High visual quality
Needs quantitative measures

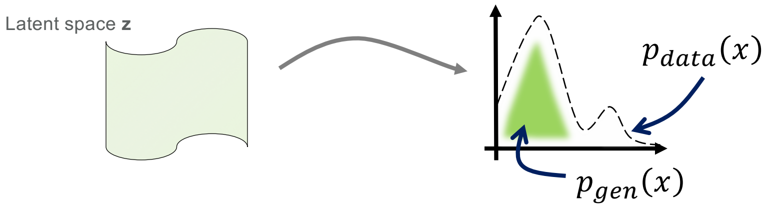
Opening the GAN black box

What does the Generator learn?



Multi-scale Representation of pore space

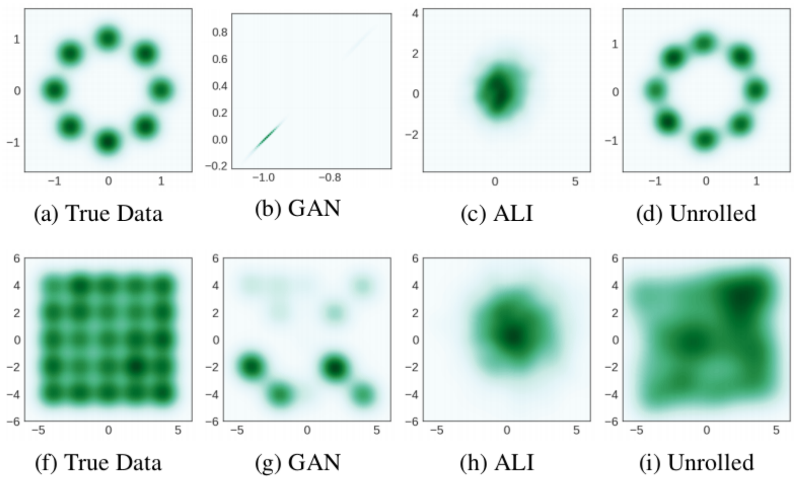
Smaller Variance in GAN generated samples: Why?



Generator can miss modes of the data distribution -> Mode-Collapse

Generative Adversarial Networks – Mode Behavior

Figure 2: Density plots of the true data and generator distributions from different GAN methods trained on mixtures of Gaussians arranged in a ring (top) or a grid (bottom).



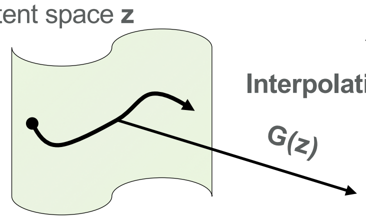
Srivastava et al.

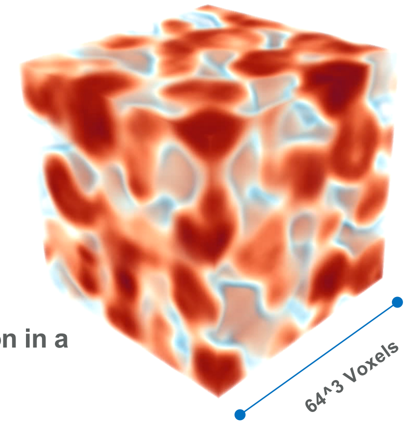
Latent Space Interpolation

Latent space **z**

$$z^* = \beta\, z_{start} + (1-\beta)\, z_{end}, \beta \in [0,1]$$

Interpolation path visualization





Interpolation in latent space:

Shows that generator has learned a meaningful representation in a lower dimensional space!

Image Inpainting (Yeh et al. 2016)

Task: Restore missing details given a corrupted / masked image $M \cdot \tilde{x}$

Use a generative model G(z) to find missing details, conditional to given information.

Contextual Loss: $L_{content} = \lambda \left|\left| M \cdot G(z) - M \cdot \tilde{x} \right|\right|_2$

Perceptual Loss: $L_{perc} = log(1 - D(G(z)))$

Optimize loss by modifying latent vector z

$M \cdot \tilde{x}$ | **Human Artist** | $L_2\ Loss$ | $L_{content} + L_{perc}$

Credit: Kyle Kastner

33

Conditioning of Generative Models

- Generative model should incorporate additional information:
  e.g. G(z, class) -> G(z, class={Cat, Dog, Bird})

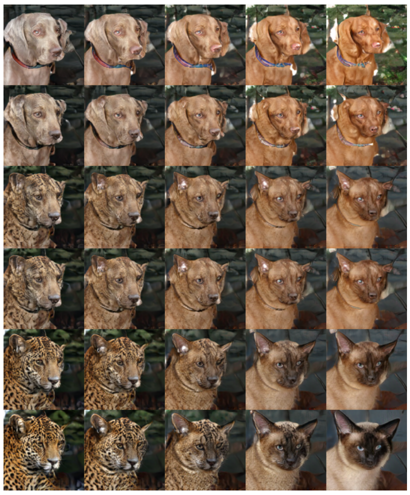
Train one generative model on images with associated class labels.

Generative model: Latent Vector + Class

Possible to perform smooth inter-class interpolation

This model -> Trained on Imagenet (1000 classes)

(Cat, Dog, Leopard, Dachshund)



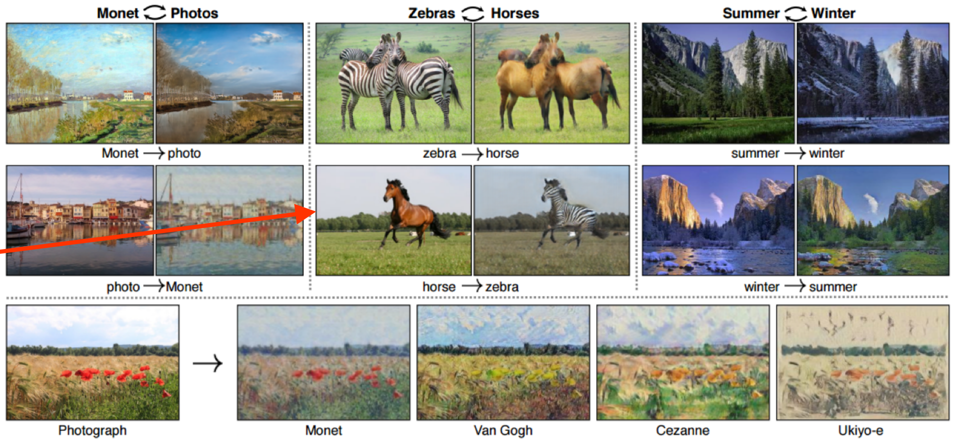
Miyato and Koyama 2018

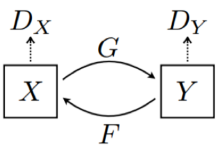
Unpaired Domain transfer - CycleGAN

Zhu et al. 2017:

Zebras ≈ Horses

But no images of one horse dressed up as a zebra exist!





(a)

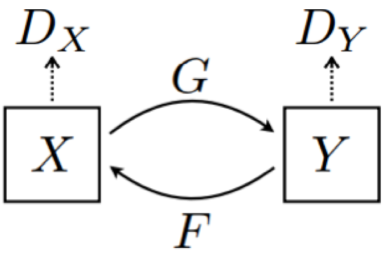
Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks

Find functions G(X) and F(Y) to map between two unpaired domains

CycleGAN – Detailed Analysis

CycleGAN consists of 4 Convolutional Neural Networks:

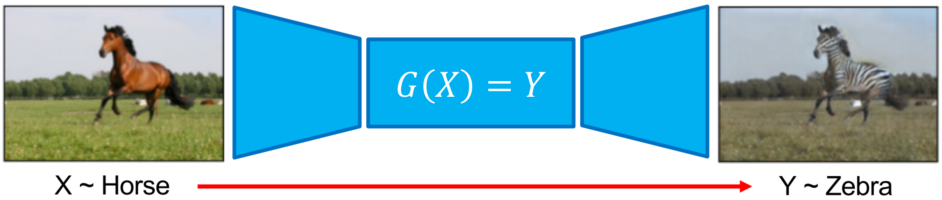
- 2 Transformer Networks: $G(X) = Y \text{ and } F(Y) = X$
- 2 Discriminators: $D_1(X) \text{ and } D_2(Y)$



(a)

Transformer Networks:

- Perform A -> B mapping, where A and B are images
  E.g.: (Cat -> Dog), (Winter -> Summer), (Blurred -> Detailed)



Comparison VAE's and GANs

VAE	GAN
• Generative + Inference Model	• Generative Model Only
• Encoder + Decoder	• Generator - Discriminator
• For new data x point provides:	• Can only sample new data
• Reconstruction x*	• NO Latent Vector z
• Latent Vector z	• NO Reconstruction x*
• Trained by maximizing ELBO	• Adversarial Training

Try out for yourself

https://poloclub.github.io/ganlab/

https://ganbreeder.app/