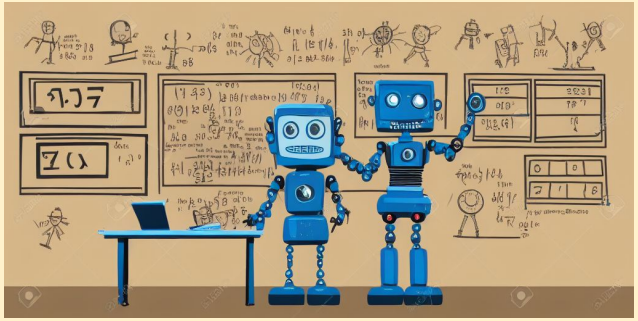


Advanced Workshop on Machine Learning

Lecture 4: Generative Adversarial Networks

Danylo Vashchilenko

2022

Agenda

Part 1

1. Generator Task
2. Discriminator Task
3. Dataset: FFHQ
4. Case Study: StyleGAN

Part 2

5. Assignment 4: FakeNet
6. Case Study: SRGAN
7. Case Study: Stable Diffusion
8. Case Study: Adversarial Latent Autoencoders

Part 1

Generator and Discriminator Tasks

(Image by a Stable Diffusion model)

Generator Task

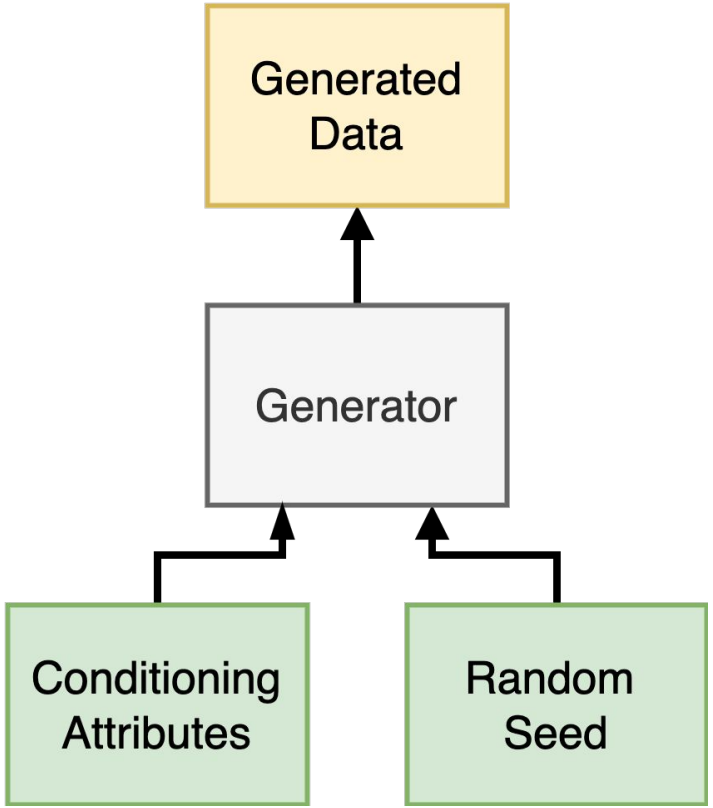
Generator learns a data distribution from the training dataset.

Generator is deterministic.

Attributes:

- Classes
- Text
- Images

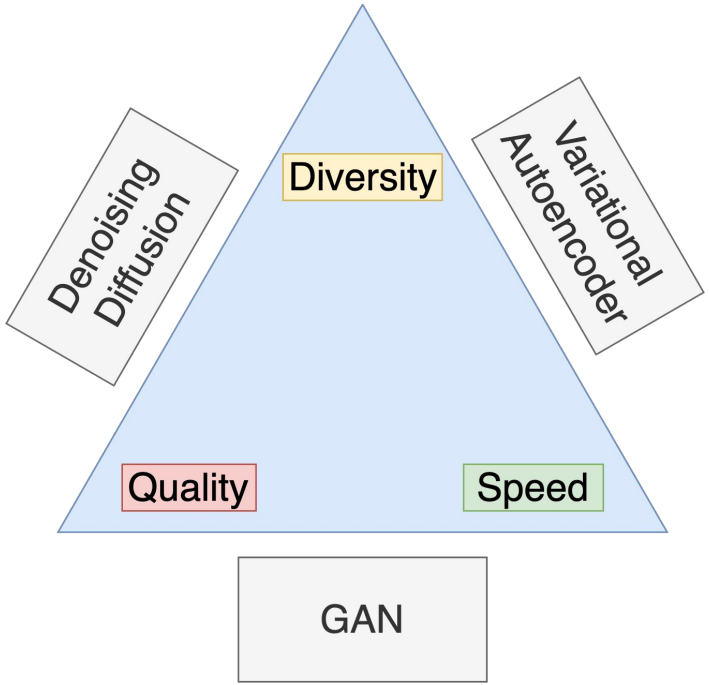
In this lecture: image examples.



Generator Architectures

Trade-offs:

- GANs lack diversity
- Diffusion Models are slow
- VAEs lack quality

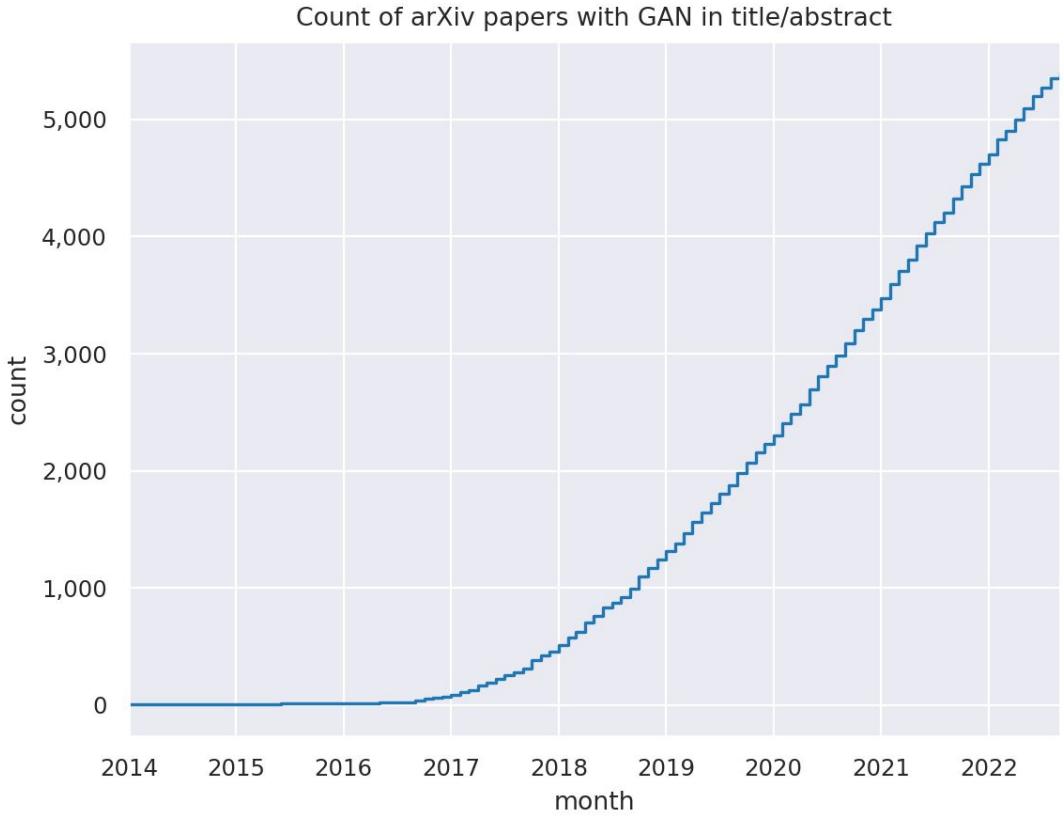


Xiao, Z., et al. (2022)

GAN Zoo

Papers on ArXiv:

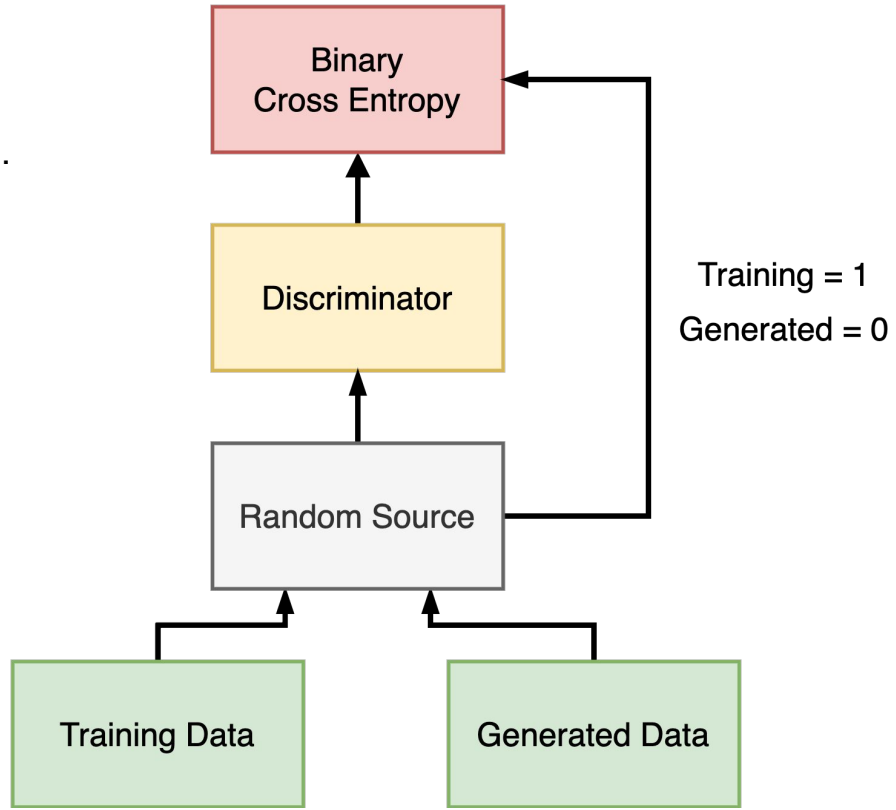
1200 new papers in 2021



See: GANArxiv.ipynb

Discriminator Task

True label depends on the source of the input data.



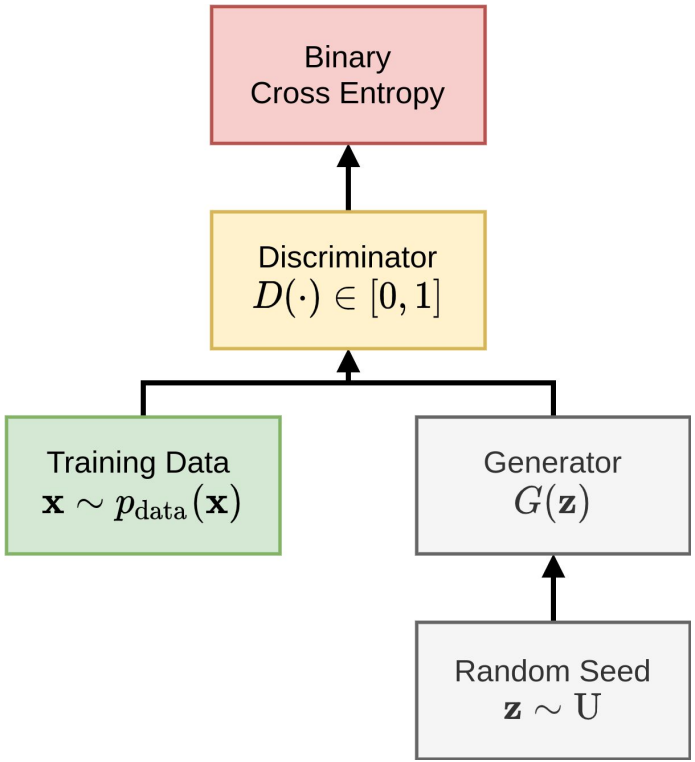
Goodfellow, I., et al. (2014)

Discriminator Training

Discriminator given a batch of examples from training data and the generator's output.

Cost is the average binary cross-entropy across the batch:

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



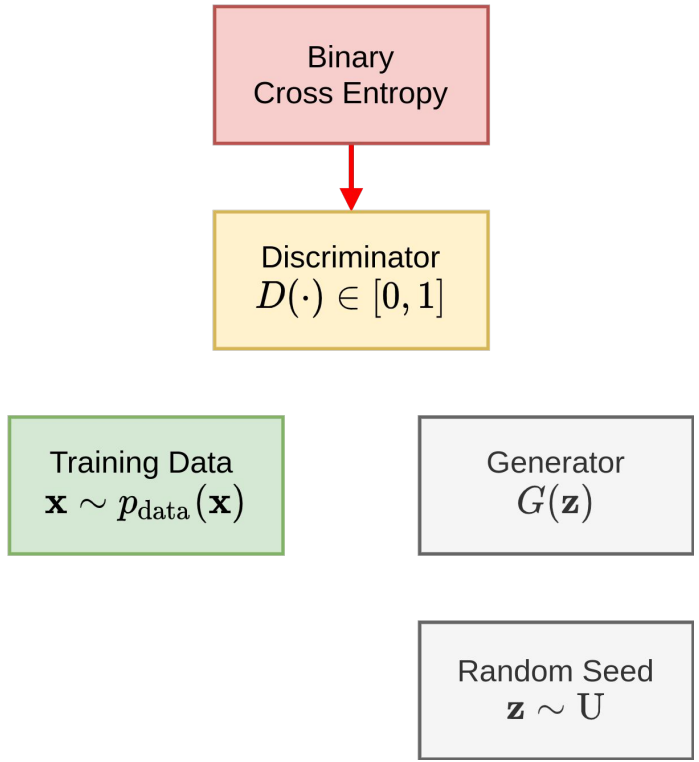
Goodfellow, I., et al. (2014)

Discriminator Training

Discriminator given a batch of examples from training data and the generator's output.

Cost is the average binary cross-entropy across the batch:

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



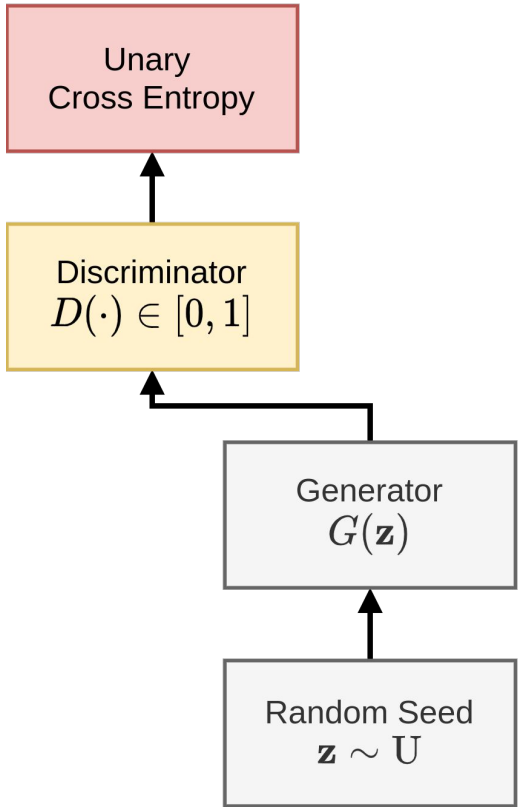
Goodfellow, I., et al. (2014)

Generator Training

The discriminator is given a batch from the generator, so the label = 0.

The generator's cost goes down when the discriminator is wrong:

$$C_G = -\frac{1}{m}\sum_{i=1}^{m}\left[\log\left(D(G(\mathbf{z}^{(i)}))\right)\right]$$



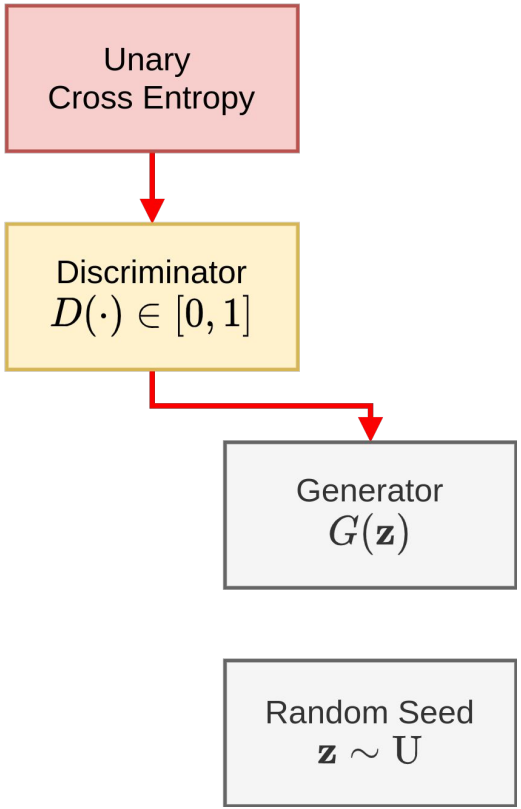
Goodfellow, I., et al. (2014)

Generator Training

The discriminator is given a batch from the generator, so the label = 0.

The generator's cost goes down when the discriminator is wrong:

$$C_G = -\frac{1}{m}\sum_{i=1}^{m}\left[\log\left(D(G(\mathbf{z}^{(i)}))\right)\right]$$



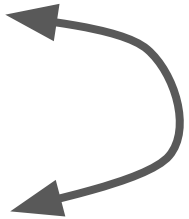
Goodfellow, I., et al. (2014)

Training Procedure

1. Sample a batch of training and generated data
2. Train the discriminator for several steps

3. Sample a batch of noise
4. Train the generator

5. Repeat steps 1-4 until convergence.



Training one component at a time

Goodfellow, I., et al. (2014)

Combined Training

$$V(D,G) = \mathop{\mathbb{E}}_{\mathbf{x}}\left[\log D(\mathbf{x})\right] + \mathop{\mathbb{E}}_{\mathbf{z}}\left[\log\left(1 - D(G(\mathbf{z}))\right)\right]$$

Rewards true positives

Rewards true negatives

$$\min_G \max_D V(D,G)$$

$$G(\mathbf{z}) \rightarrow p_{\text{data}}(\mathbf{x})$$

$$D(G(\mathbf{z})) \rightarrow 0.5$$

Goodfellow, I., et al. (2014)

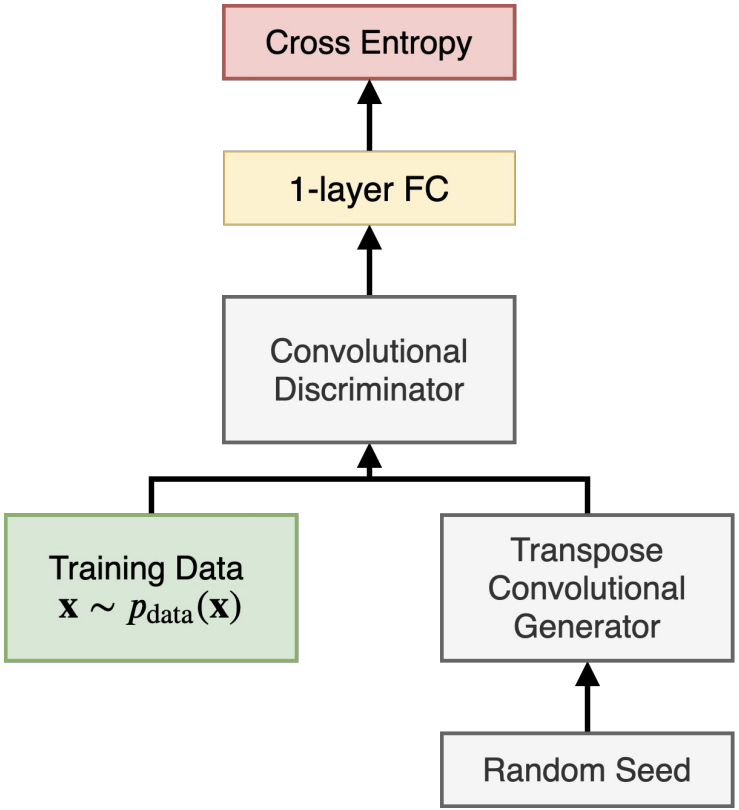
Case Study: Deep Convolutional GAN

Convolutional Discriminator

Regular convolutions

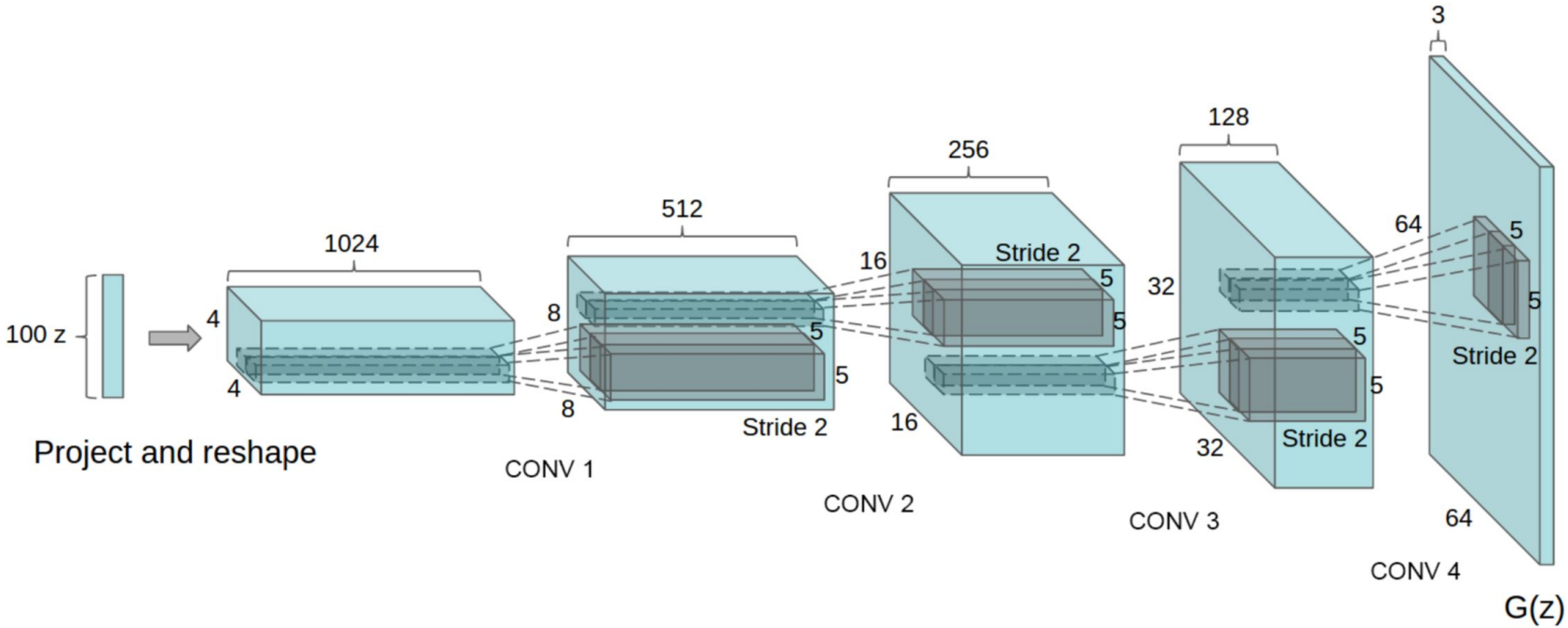
Convolutional Generator

Transpose convolutions



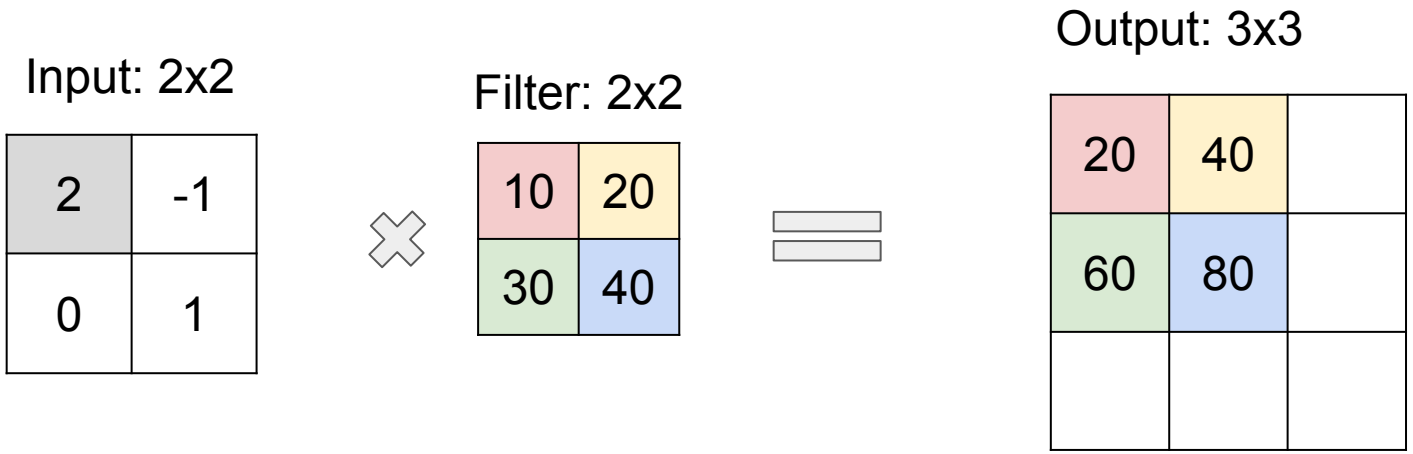
Radford, A., et al. (2016)

DCGAN: Generator

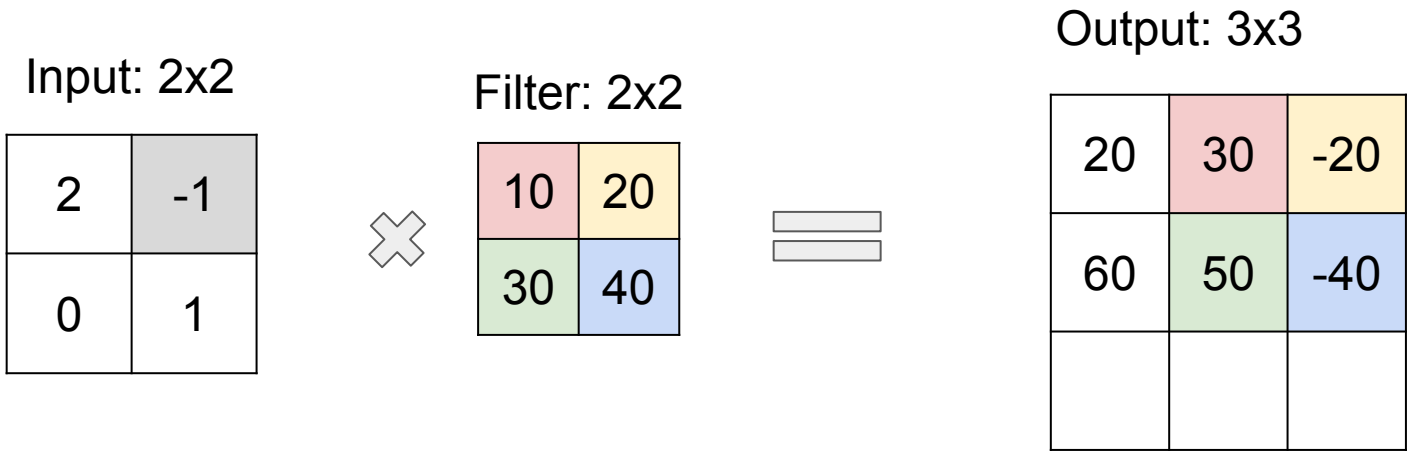


Radford, A., et al. (2016)

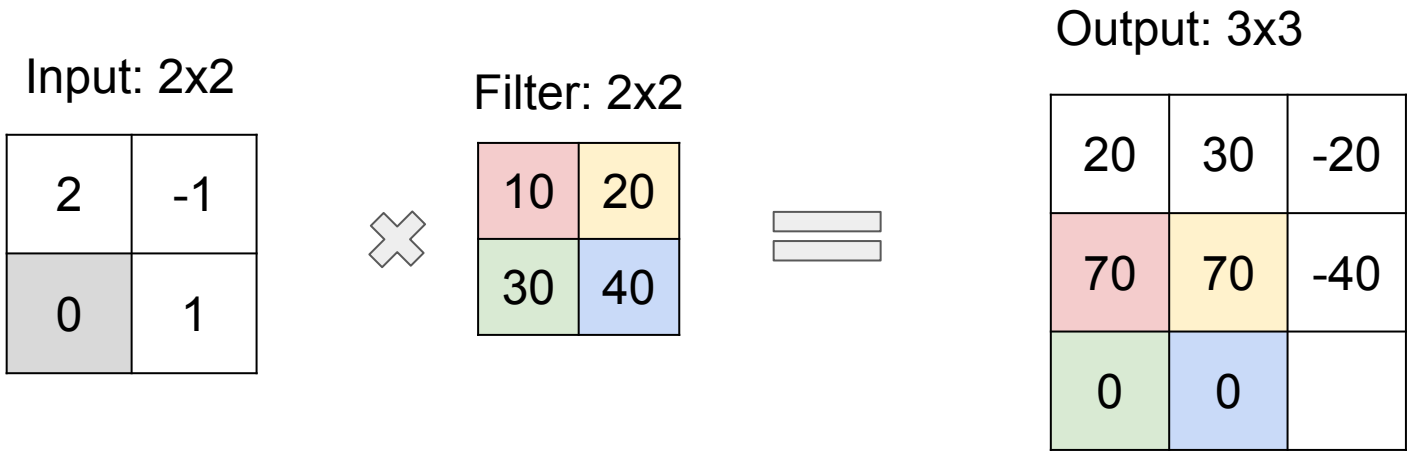
Transpose Convolution



Transpose Convolution



Transpose Convolution



Transpose Convolution

Input: 2x2

2	-1
0	1

Filter: 2x2

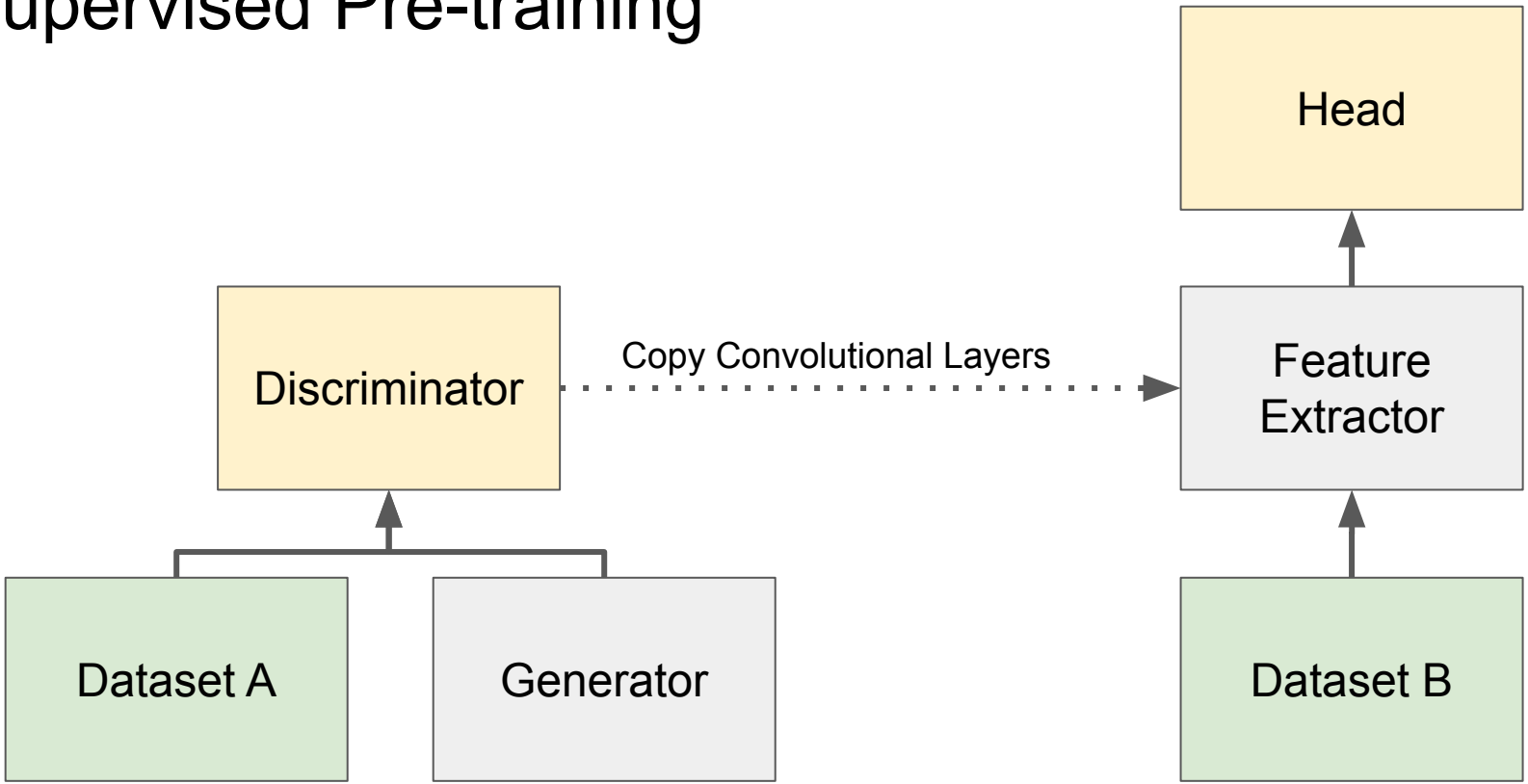
10	20
30	40

Output: 3x3

20	30	-20
70	80	-30
0	30	40

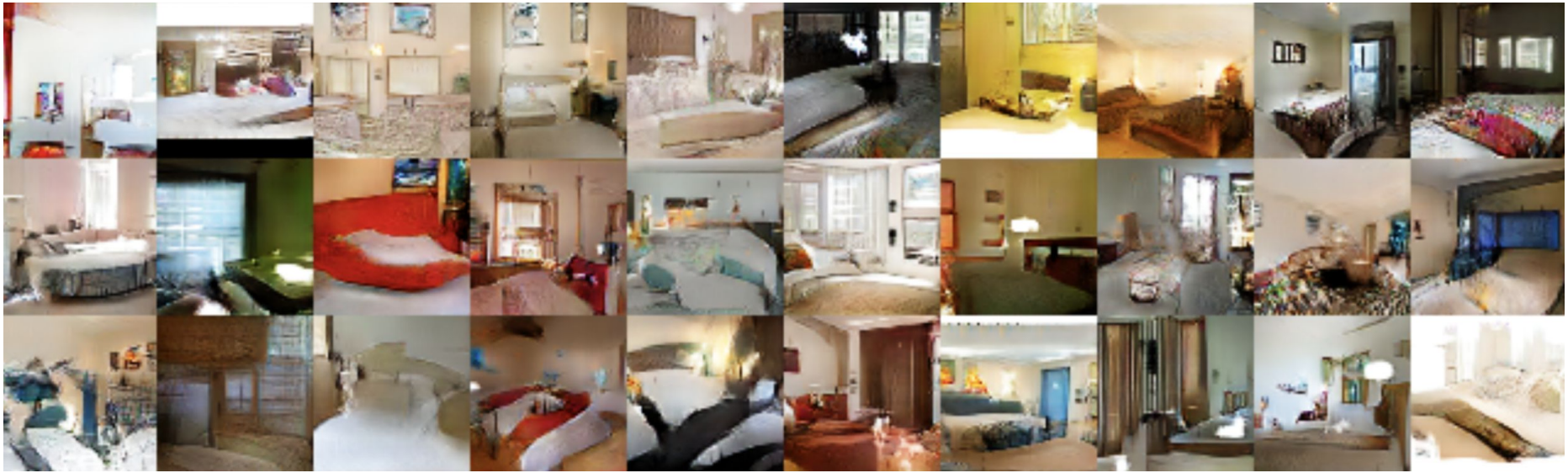
Also called fractionally-strided convolution.
Linear algebra details: https://arxiv.org/pdf/1603.07285.pdf

Unsupervised Pre-training



Radford, A., et al. (2016)

DCGAN: Generated Samples



Trained on LSUN bedroom images.

Radford, A., et al. (2016)

DCGAN: Latent Space

Interpolation between **z** vectors with and without windows in the generated image.

Radford, A., et al. (2016)

Selected Models and Examples

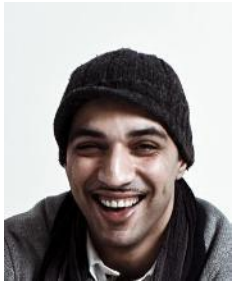
BigGAN - Brock, A., et al. (2019)

StyleGAN2 - Karras et al (2019)

Ryan Hover

https://arthurfindelair.com/thisnightskydoesnotexist/

Dataset: CelebFaces Attributes



Dimensions
200K images
20 attributes
178 x 178

Open Source
http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

Source: Deep Learning Face Attributes in the Wild (Ziwei, L., et al., 2015)

Dataset: Flickr-Faces HQ (FFHQ)

Dimensions
70K images
1024 x 1024

Github
https://github.com/NVlabs/ffhq-dataset

Source: Karras, T., et al. (2018)

NVIDIA Research Lab

Models

ProgressiveGAN, 2018
StyleGAN, 2019
StyleGAN V2, 2020
StyleGAN V2 - ADA, 2020
StyleGAN V3, 2021

Datasets

- Flick Faces HQ
- CelebA HQ

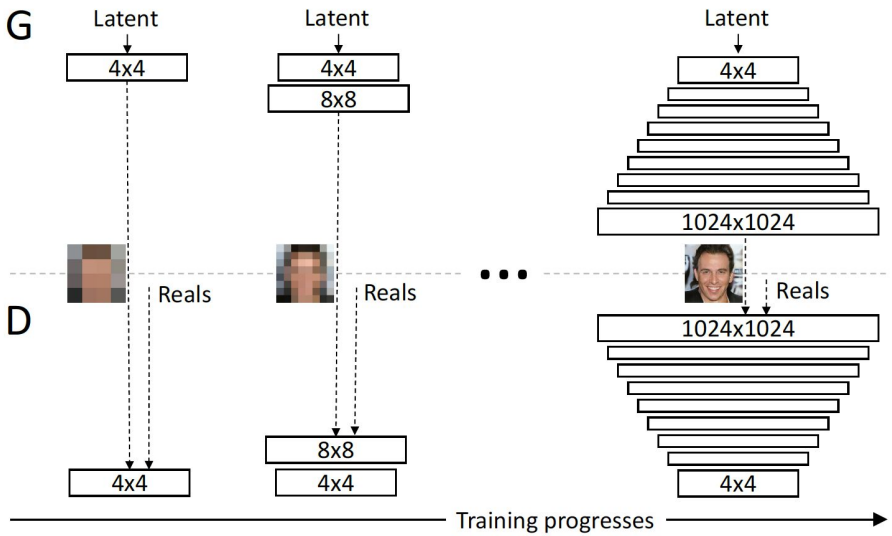
Case Study: StyleGAN

NVIDIA Research Labs, 2020

Available pre-trained on:

- Faces: FFHQ
- Animals: AFHQ
- CIFAR10

https://github.com/NVlabs/stylegan3



(Karras, T., et al., 2018)

Source: Training Generative Adversarial Networks with Limited Data (Karras, T., et al., 2020)

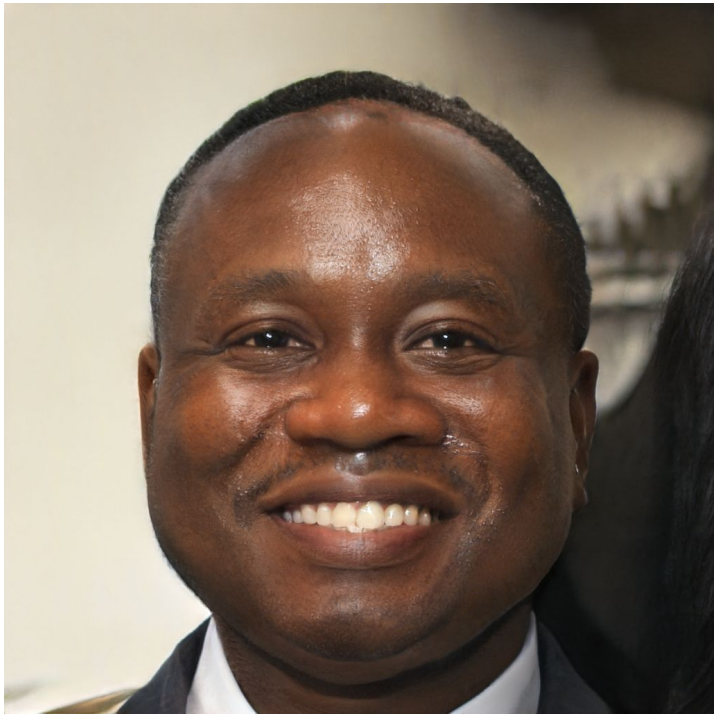
StyleGAN: Examples

Source: Training Generative Adversarial Networks with Limited Data (Karras, T., et al., 2020)

StyleGAN2: Examples

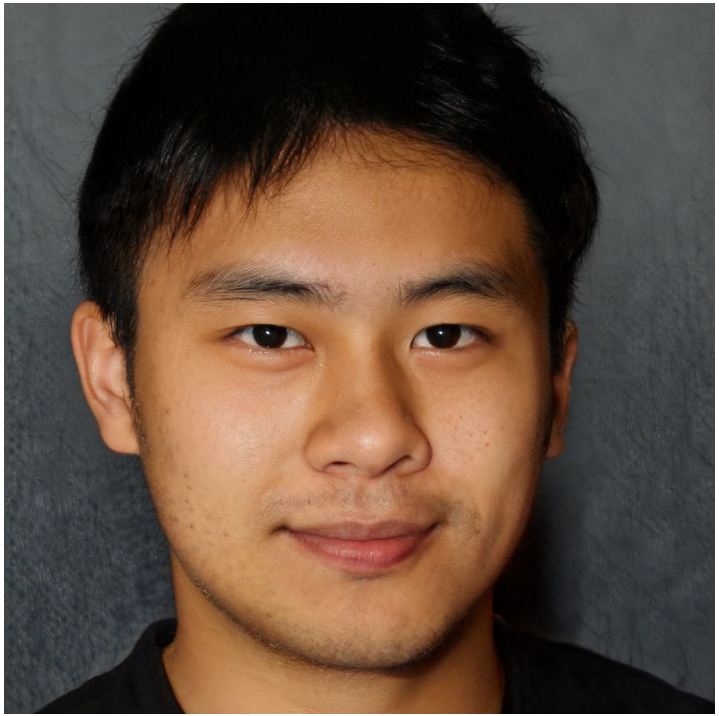
Source: Training Generative Adversarial Networks with Limited Data (Karras, T., et al., 2020)

StyleGAN2: Examples



Source: Training Generative Adversarial Networks with Limited Data (Karras, T., et al., 2020)

StyleGAN2: Examples



Source: Training Generative Adversarial Networks with Limited Data (Karras, T., et al., 2020)

Part 2

FakeNet and Other Models

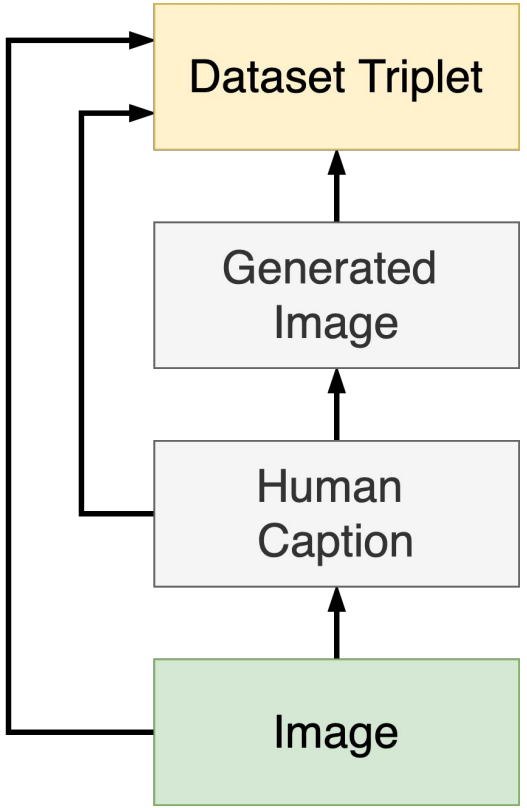
(Image by a Stable Diffusion model)

Dataset: FakeNet

Dimensions
224 x 224 x 3
~12K triplets

Triplets
Original image
English caption by human
Generated image from caption

Task
Determine which 1 of 2 images is generated



FakeNet: Examples

A pile of oranges sitting on top of each other.

Original

Generated

FakeNet: Examples

A close of a small bird perched on a tree limb.

Original

Generated

FakeNet: Examples

The Big Ben clock tower sitting under a dark sky.

Original

Generated

FakeNet: Examples

Small bathroom with red curtains on the outside of the shower.

Original

Generated

FakeNet: Examples

Four brown bears are walking through a stream.

Original

Generated

FakeNet: Examples

The cat is sleeping underneath the clock.

Original

Generated

FakeNet: Examples

a man plays baseball on a field with grass.

Original

Generated

Baseline Models

Note
Very simple architectures
All models trained to convergence
No regularization
No fine-tuning

Other considerations
Development time
Training time
Parameter count

backbone	input_type	binary accuracy (t=0.5) on score
vanilla cnn	single image	0.9398
	paired images	0.8967
mobile net	paired images	0.7532
	single image	0.7427

Accuracy = 0.70 is a reasonable baseline.

Demo

AssignmentTemplate.ipynb

Assignment 4

1. Grading:
   a. 10 points if Accuracy (t = 0.5) on Score > 0.70
   b. 10 points proportional to the percentile in class
2. Discriminate between real and fake images
3. Deliverables:
   a. Jupyter notebook
   b. Saved model definition
   c. Saved model parameters
   d. Predicted labels for the Score segment

4. Due by EOD 10/23

Case Study: SuperResolution GAN

Supervised Task

Given a low-resolution image, reconstruct the high-resolution image.

Prior Art

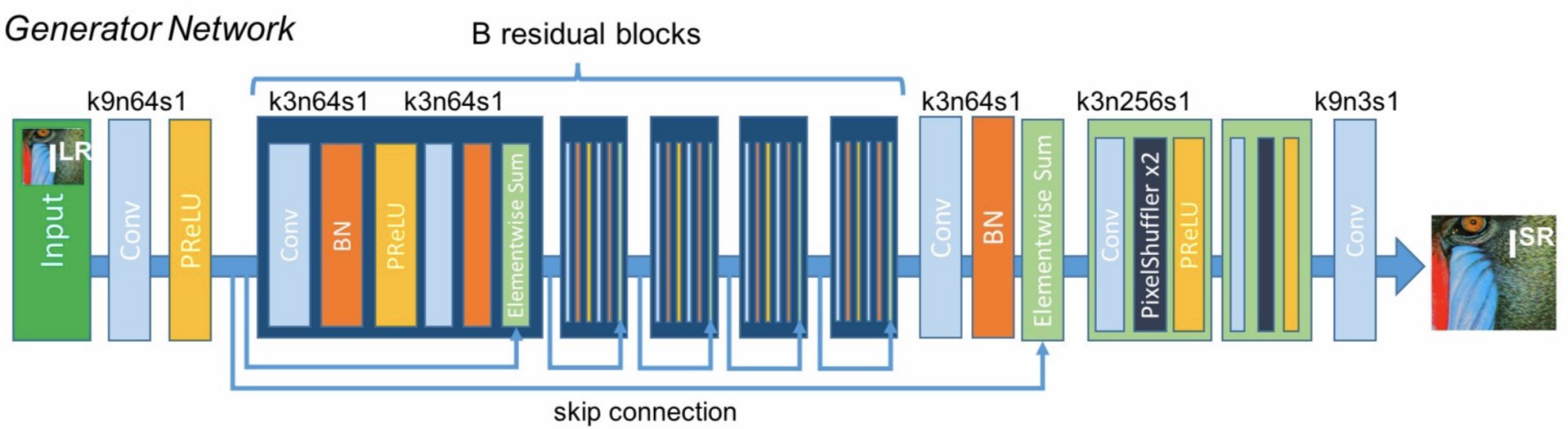
Residual Transpose CNN with MSE loss has shown great performance on this task but the reconstructions are blurry.

Bicubic

Original

Ledig, C., et al. (2017)

SRGAN: Generator CNN



Similar to ResNet, but uses Transposed Convolutions to increase the spatial dimensions.

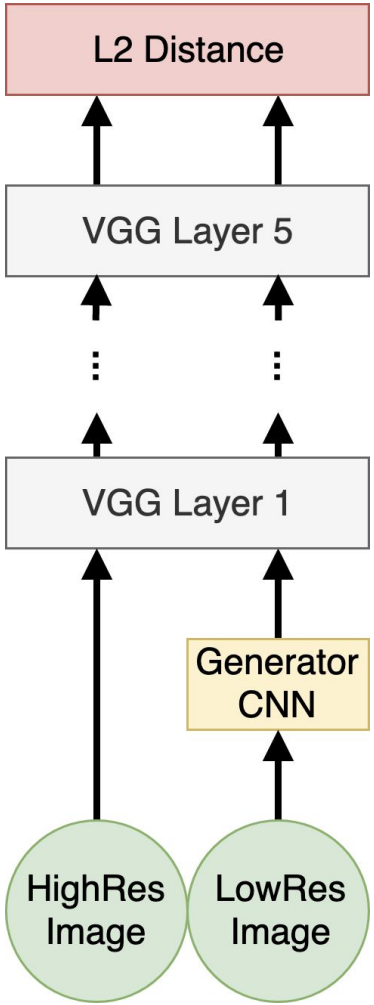
Ledig, C., et al. (2017)

SRGAN: VGG Loss

Comparison of reconstructed vs the original image is done using pre-trained VGG19.

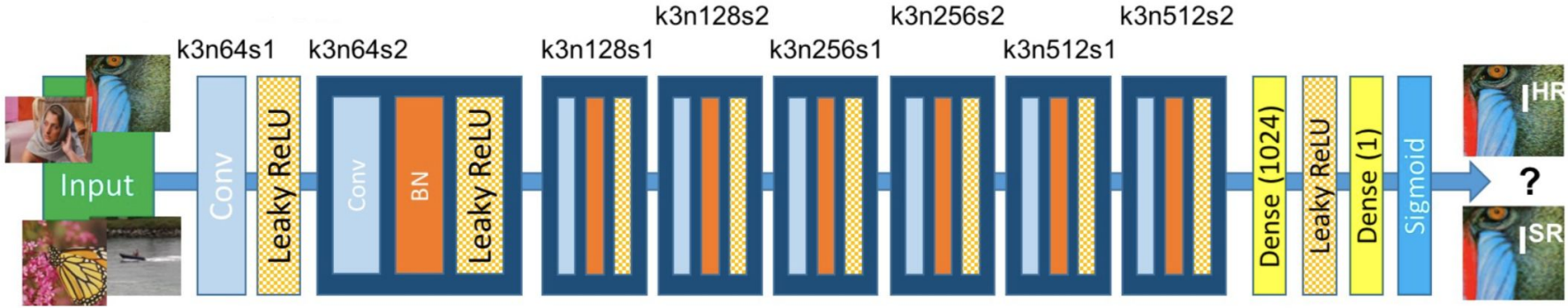
Both images go through several convolutional layers of VGG19, and the final activation volumes are compared.

The loss is the L2 distance between the activation volumes of the reconstruction vs the original image.



Ledig, C., et al. (2017)

SRGAN: Discriminator CNN



1. Fully-convolutional CNN
2. MLP Classifier
3. Binary Cross Entropy.

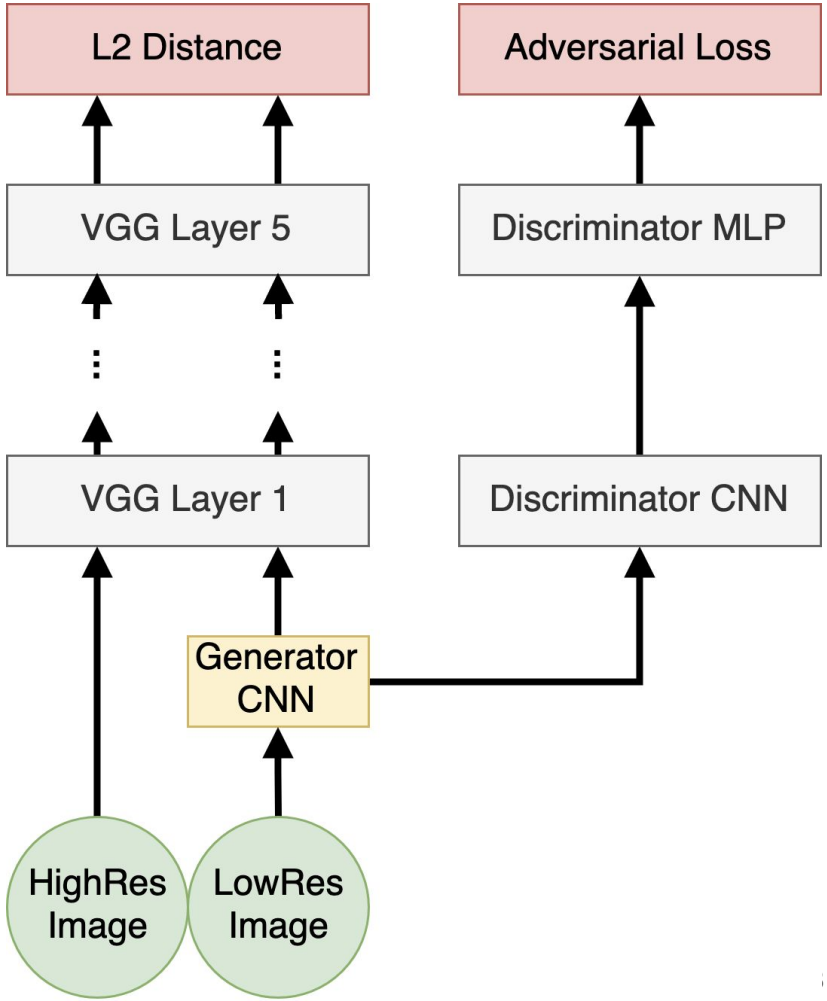
Ledig, C., et al. (2017)

SRGAN: Forward Pass

Dual Loss

Weighted sum of Adversarial Loss and VGG Activation Loss

Authors find the AL+VGG combo to work better than individual losses or MSE.



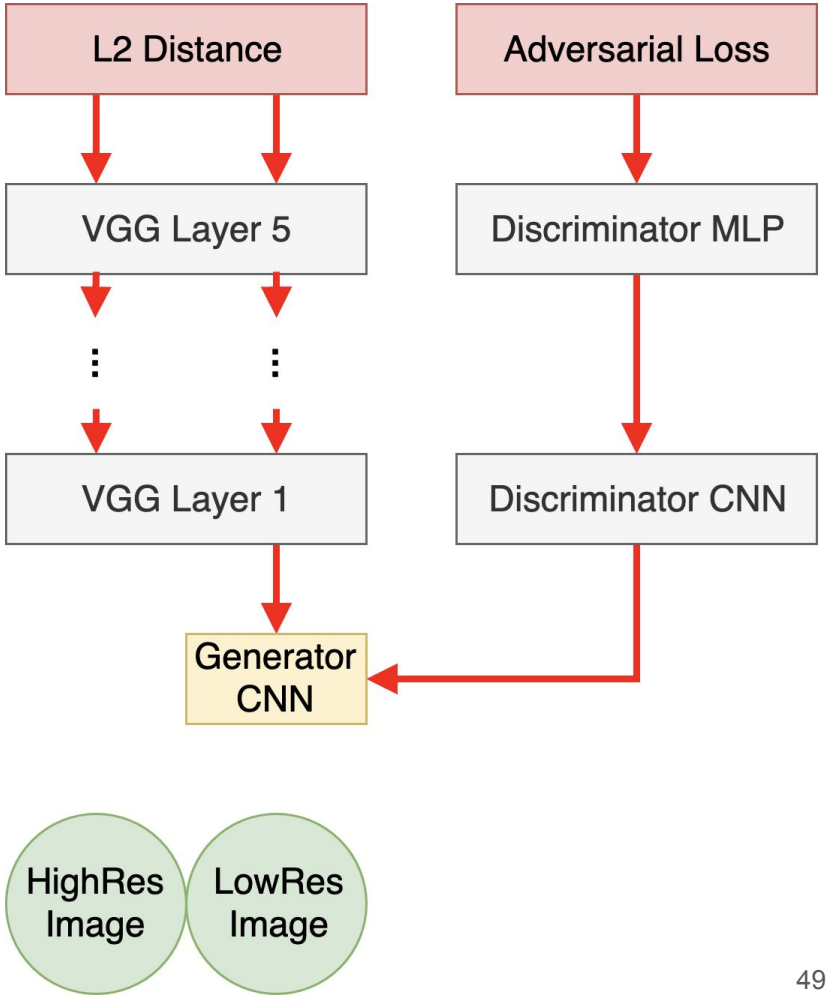
Ledig, C., et al. (2017)

SRGAN: Backprop

Dual Loss

Weighted sum of Adversarial Loss and VGG Activation Loss

Authors find the AL+VGG combo to work better than individual losses or MSE.



Ledig, C., et al. (2017)

SRGAN: Examples

Bicubic

MSE

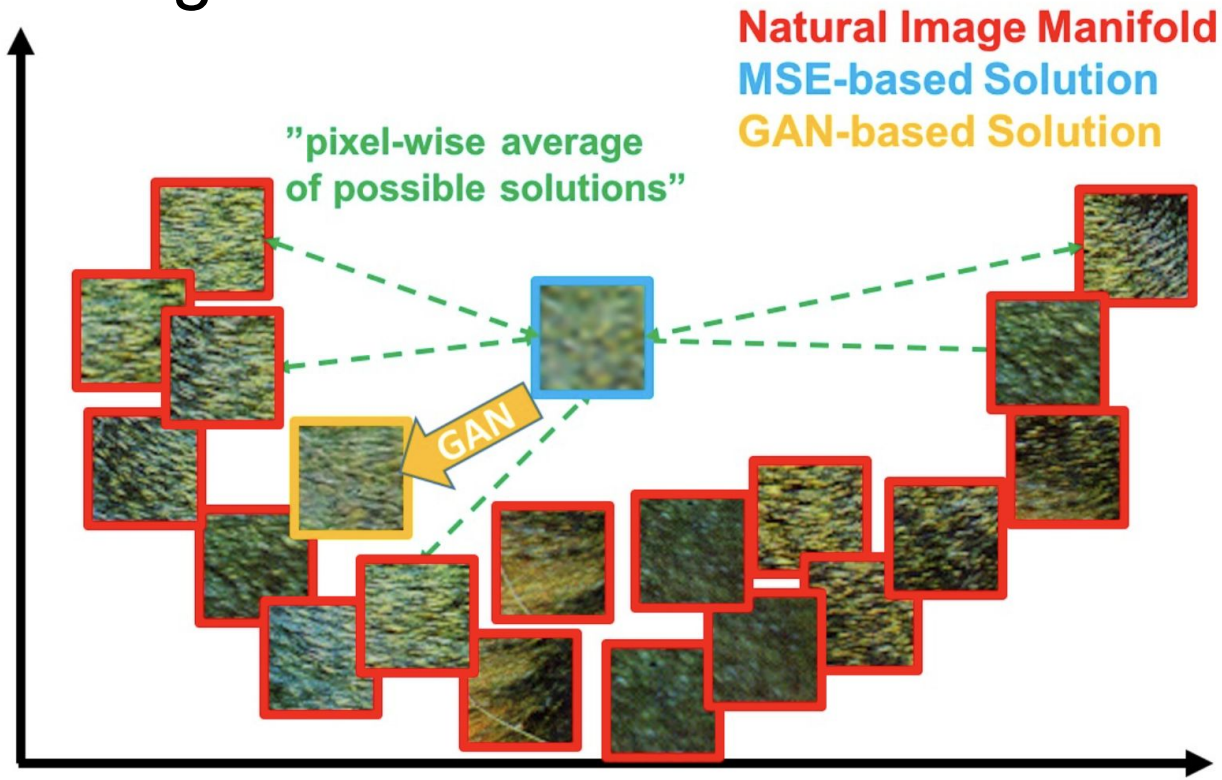
VGG+AL

Original

Ledig, C., et al. (2017)

SRGAN: What's wrong with MSE?

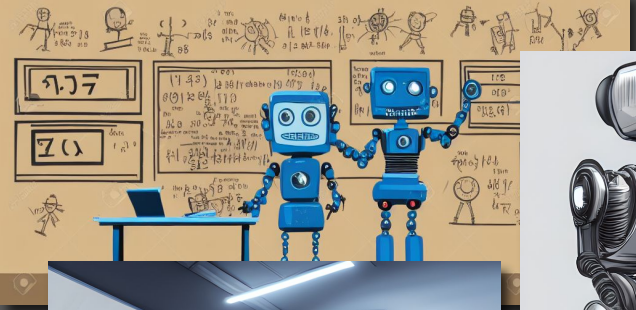
A mean is often a bad approximator of any of the samples from multi-modal distributions.

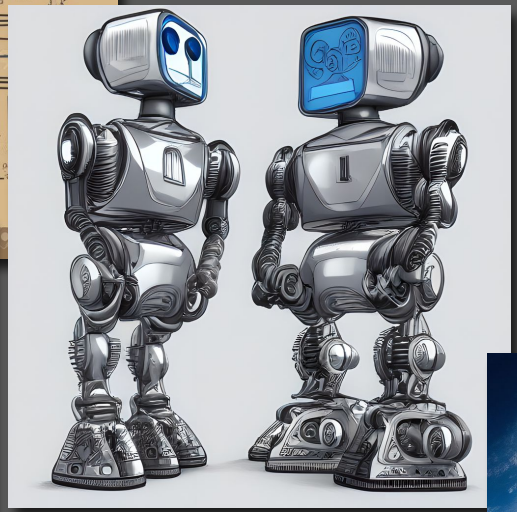


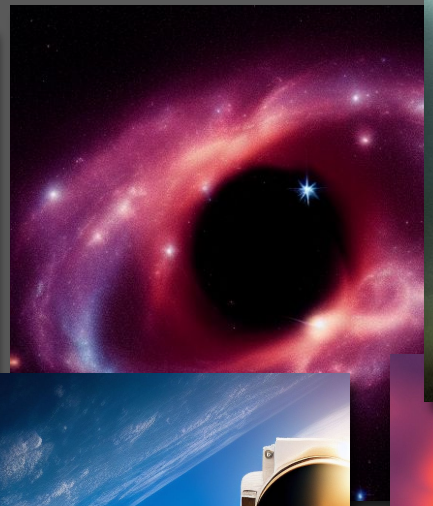
Ledig, C., et al. (2017)

Demo

StableDiffusion.ipynb







Case Study: BigGAN

Which one of these images is real?

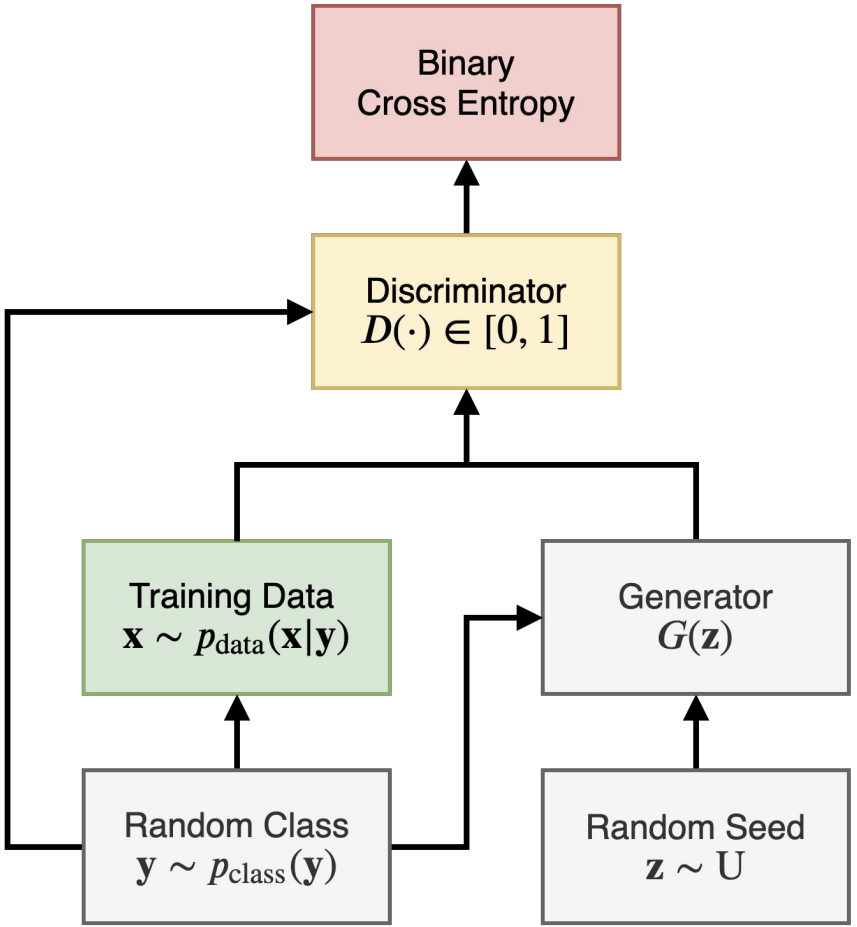
Brock, A., et al. (2019) / USDA (2014)

BigGAN

Class-conditional generation

High-resolution images
512 x 512

Quality vs diversity trade-off
via noise truncation



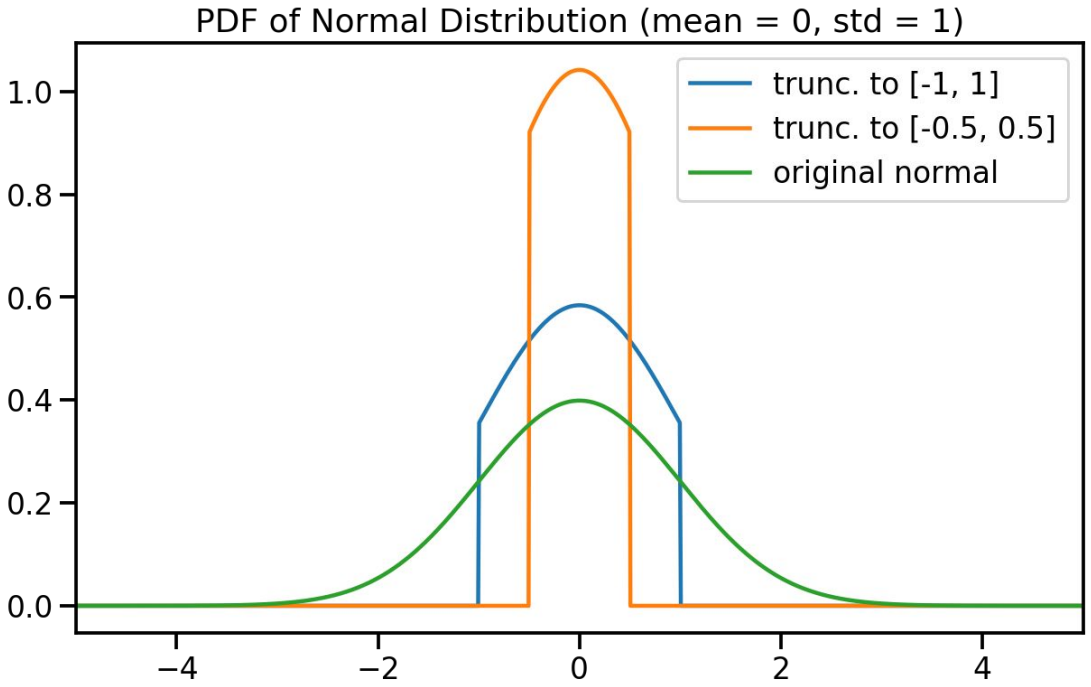
Brock, A., et al. (2019)

BigGAN: Noise Truncation

Properties of symmetrically truncated normal distribution:

1. excludes low-probability events
2. preserves mean
3. lowers variance

Enables trade-off of diversity for quality.



Brock, A., et al. (2019)

BigGAN: Diversity

Larger noise space allows to generate diverse sets of same-class images.

Since low-probability noise was less frequently checked by the discriminator, its quality is lower.

Diversity and quality are a trade off.

Brock, A., et al. (2019)

BigGAN: Class Interpolation

1. Randomly choose a noise vector

2. Choose two classes
   (terrier and tiger in this example)

3. Linearly interpolate
   one-hot-encoded class vectors

4. Feed the constant-noise vector, and
   the class vectors to the generator.

Brock, A., et al. (2019)

iPod to ice cream

Train to golf cart

Demo

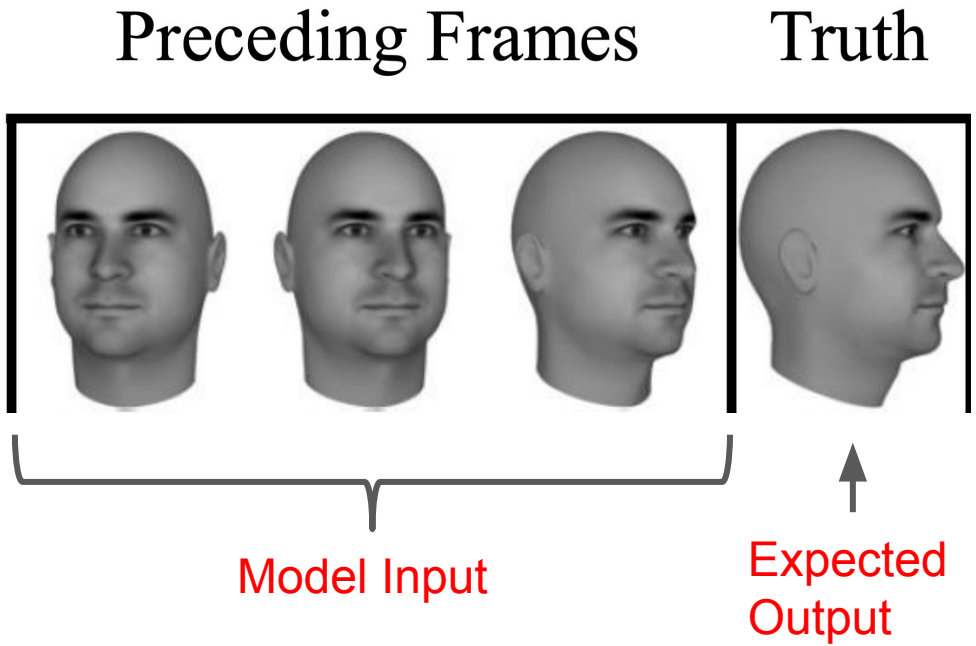
BigGAN.ipynb

Case Study: Predictive Generative Network

Task
Predict the next frame in a sequence given last 3 frames

Every sequence has a randomly generated face rotating around Y axis.

Initial angle and angular velocity are random across sequences.



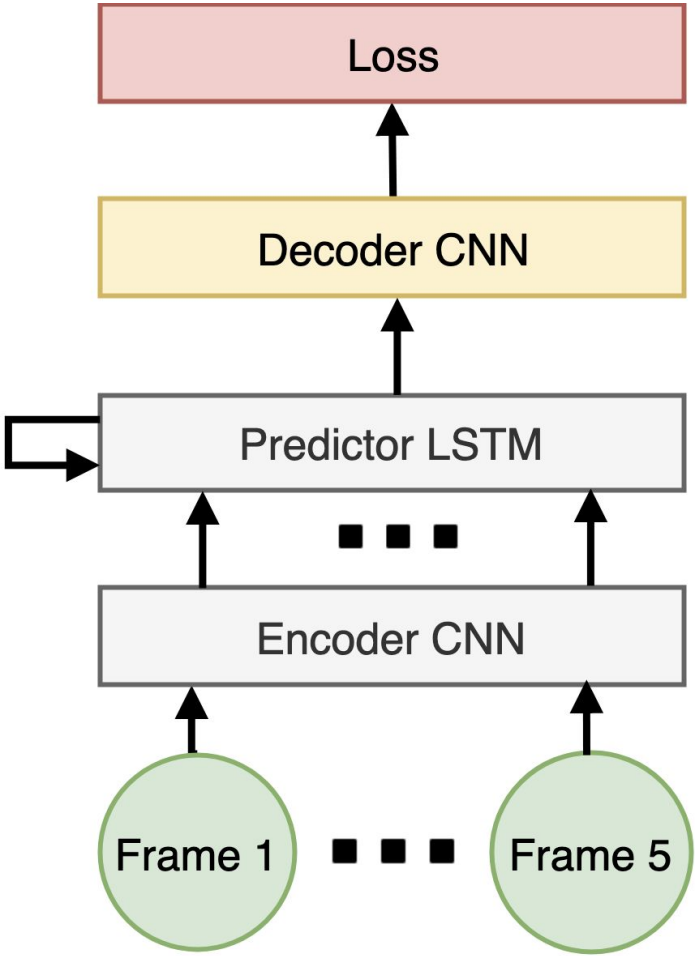
Lotter, W., et al. (2016)

PGN: Generator Model

1. Encoder CNN extracts each frame's features into a low-dimensional encoding.

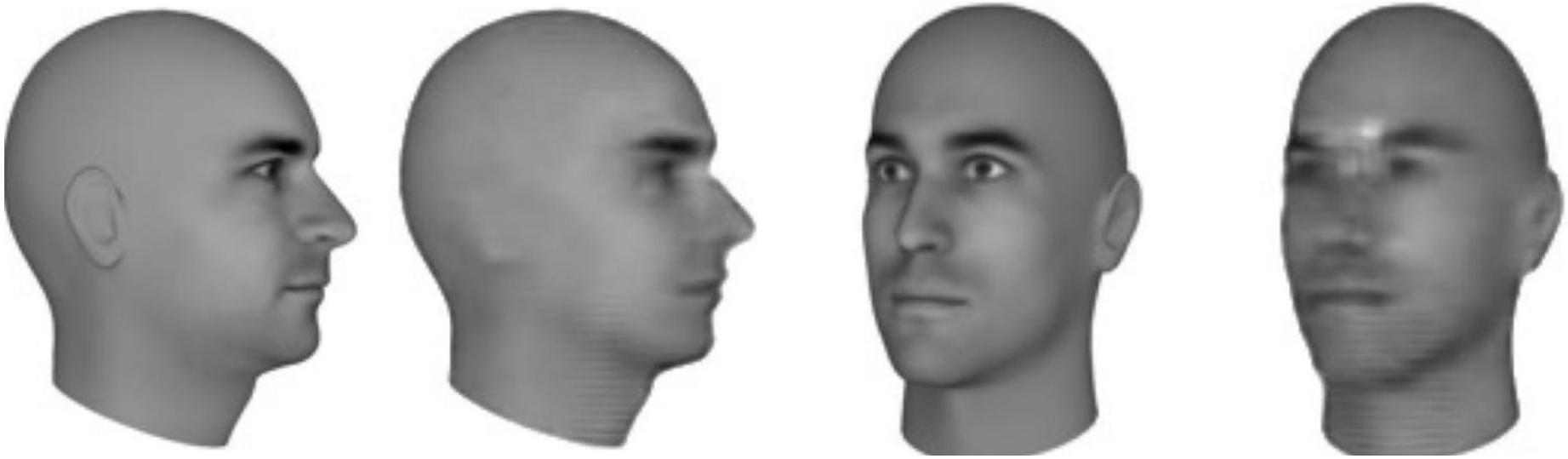
2. Predictor LSTM consumes each encoding sequentially, and produces an encoding for the target frame.

3. Decoder CNN converts the encoding into the target frame.



Lotter, W., et al. (2016)

PGN: Mean Squared Error



Target Prediction Target Prediction

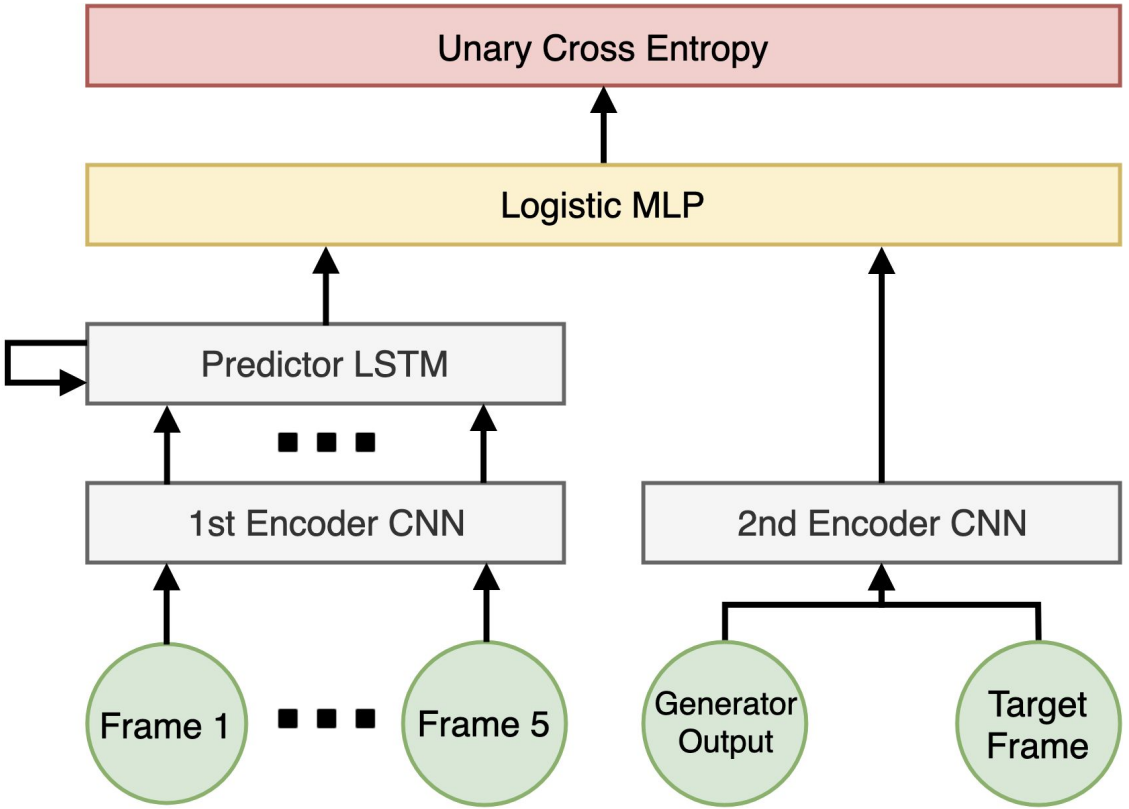
Lotter, W., et al. (2016)

PGN: Discriminator

1st Encoder CNN and Predictor LSTM form conditional expectations about the target frame.

2nd Encoder CNN extracts features from the adversarial input.

Logistic MLP determines whether the adversarial input is the true target frame.



Lotter, W., et al. (2016)

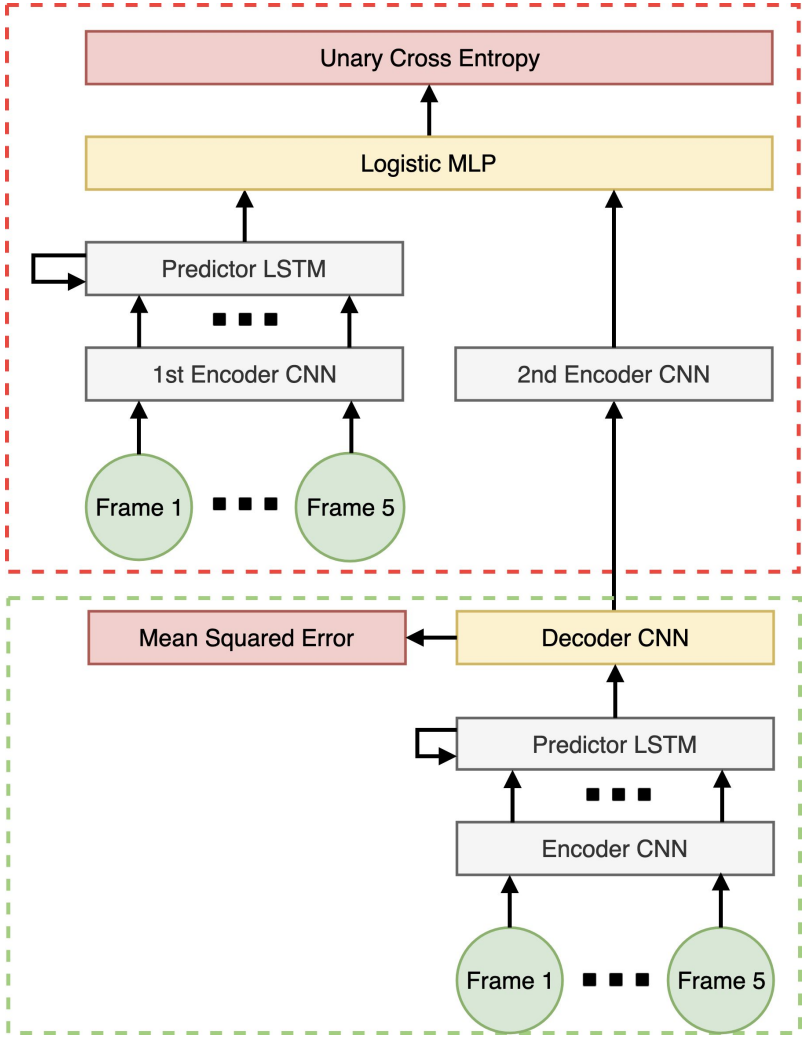
PGN: Generator Training

The generator is trained with dual loss:

- MSE against the target frame
- Adversarial loss

The final cost is a weighted sum of the two losses. Weights are hyperparams.

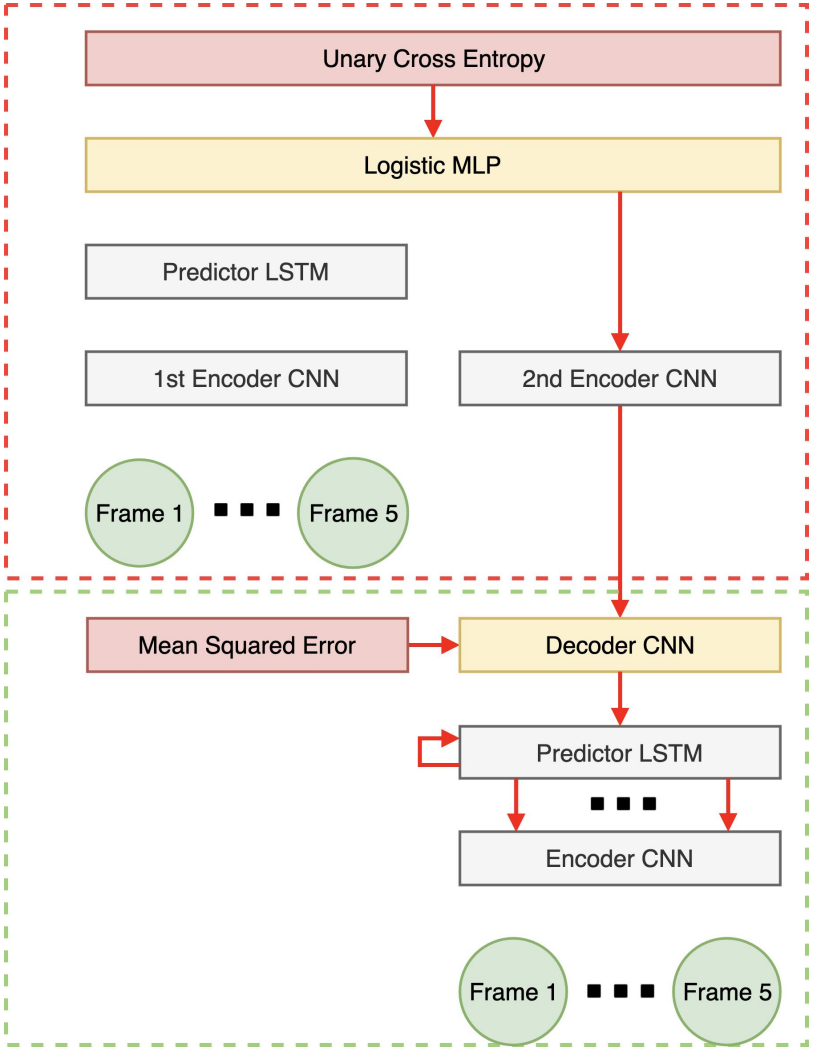
None of the LSTMs or CNNs share parameters across layers.



Lotter, W., et al. (2016)

PGN: Generator Backprop

During backward propagation gradient flows through only a part of the discriminator.

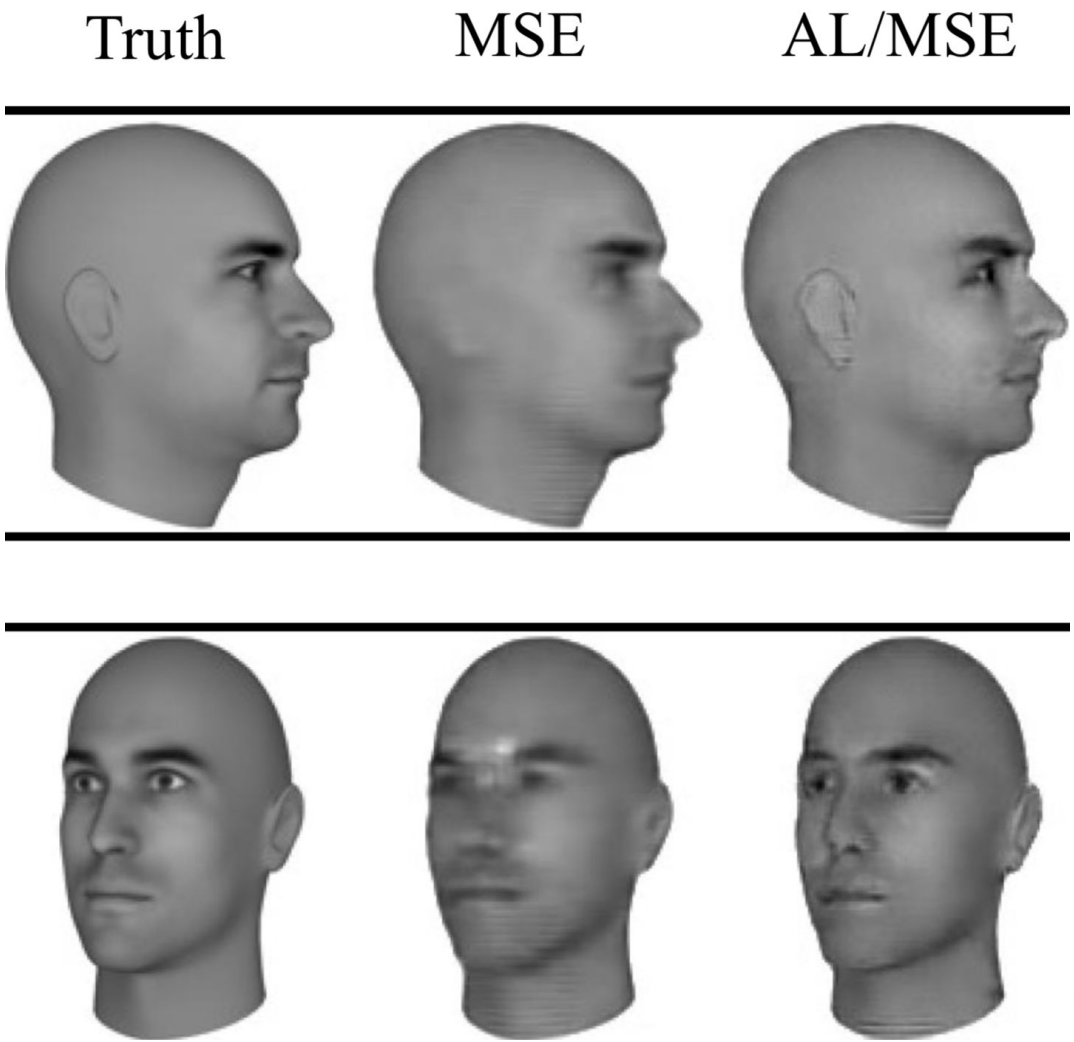


Lotter, W., et al. (2016)

PGN: Dual Loss

Model trained with Dual Loss outputs sharp frames which more closely resemble the target frames.

Read the paper for more predictive task examples.



Lotter, W., et al. (2016)

Demo
Adversarial Latent Autoencoders

References

1. A Style-Based Generator Architecture for Generative Adversarial Networks (Karras, T., et al., 2018)
2. Deep Learning Face Attributes in the Wild (Ziwei, L., et al., 2015)
3. Generative Adversarial Nets (Goodfellow, I., et al. 2014)
4. Generative Adversarial Text to Image Synthesis (Redd, S., et al., 2016)
5. Large Scale GAN Training for High Fidelity Natural Images Synthesis (Brock, A., et al., 2019)
6. NIPS 2016 Tutorial: Generative Adversarial Networks (Goodfellow, I., 2016)
7. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network (Ledig, C., et al., 2017)
8. Progressive Growing of GANs for Improved Quality, Stability, and Variation (Karras, T., et al., 2018)
9. StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks (Zhang, H., et al., 2017)
10. Training Generative Adversarial Networks with Limited Data (Karras, T., et al., 2020)
11. Unsupervised Learning of Visual Structure using Predictive Generative Networks (Lotter, W., et al., 2016)
12. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks (Radford, A., Metz, L., 2016)

Thank you

Final Q&A Time

(Image by a Stable Diffusion model)