Cornell Bowers C·IS
College of Computing and Information Science

Deep Learning

Week 6: Image-to-Image Models/ GANs

Cornell Bowers C·IS

For paired data, how can we train a model to…

Map from aerial photographs

Cornell Bowers C·IS

For paired data, how can we train a model to…

Image Super-resolution

Saharia, Chitwan, et al. "Image super-resolution via iterative refinement." IEEE transactions on pattern analysis and machine intelligence 45.4 (2022): 4713-4726.

Cornell Bowers CIS

For paired data, how can we train a model to…

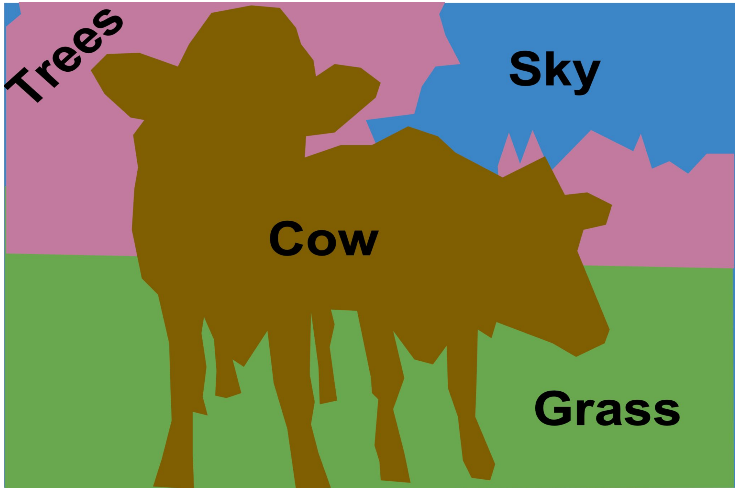


Image Segmentation

Image Credit: Stanford CS231n, Lecture 11

Review: Image Classification

Input Image

Classification

Cow

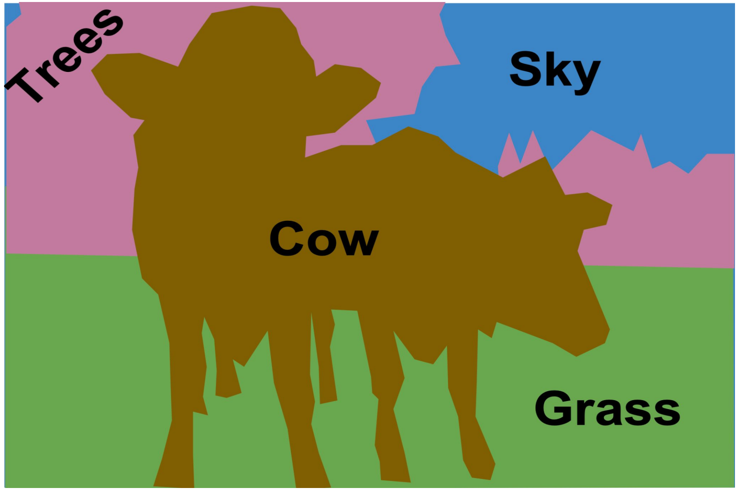
Image-level Prediction

Image Credit: Stanford CS231n, Lecture 11

Image-to-Image Task

Input Image

Semantic Segmentation



Pixel-level Prediction

Image Credit: Stanford CS231n, Lecture 11

Applications in Autonomous Driving

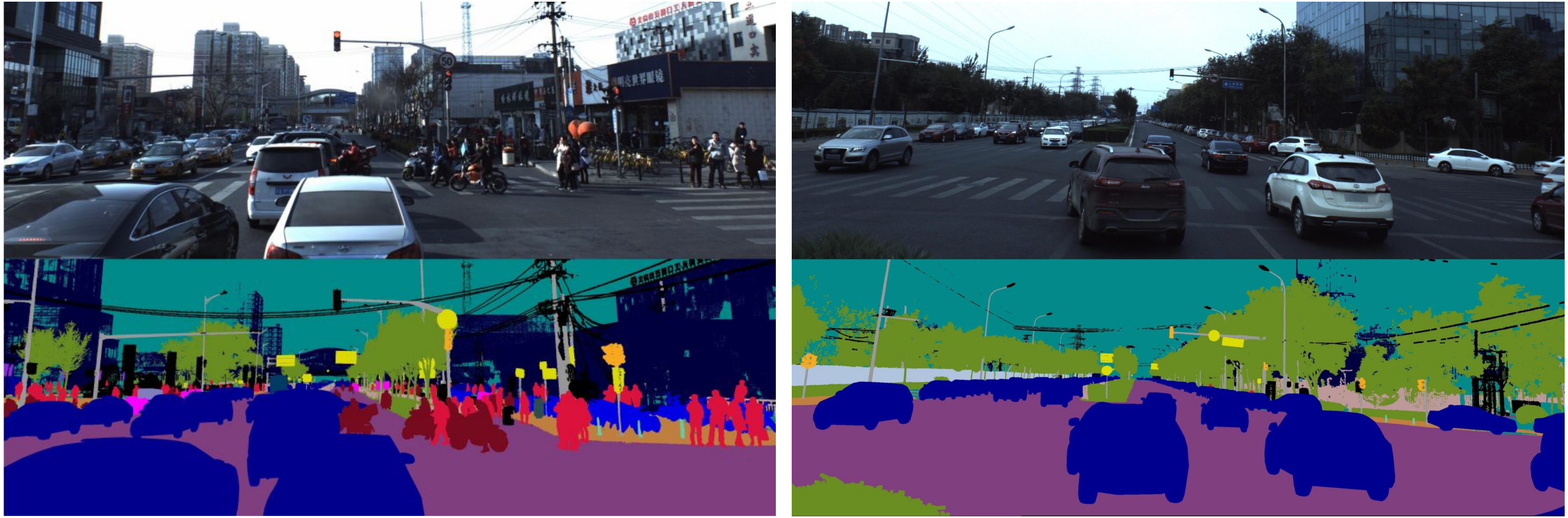


Image Credit: CVPR 2018 WAD Video Segmentation

Cornell Bowers C·IS

Applications in Medical Imaging

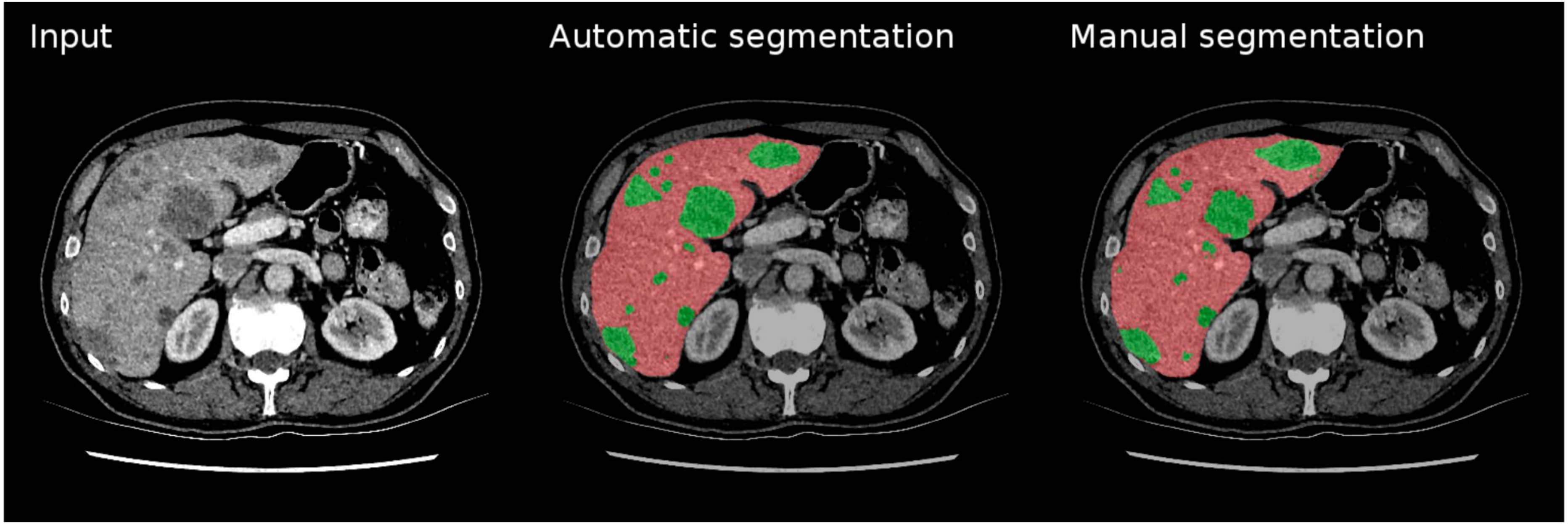


Image Credit: LiTS17 (Liver Tumor Segmentation Benchmark)

Cornell Bowers C·IS

Semantic Segmentation

Task Formulation

Take an *image* of dimension (H, W, 3) and output a *segmentation map* of dimension (H, W, 1).

Task Formulation

Take an *image* of dimension (H, W, 3) and output a *segmentation map* of dimension (H, W, 1).

Formally, it is a function f, parameterized by θ, that produces a segmentation map of C classes.

$$f_\theta : \mathbb{R}^{H \times W \times 3} \longrightarrow \mathbb{N}^{H \times W \times 1}$$

$$\xrightarrow{f_\theta}$$

1: cow
2: grass
3: tree
4: sky

Cornell Bowers C·IS

Image-to-Image Generation

A *segmentation map* of dimension (H, W, 1) can be viewed as a *generated image*.

$$f_\theta : \mathbb{R}^{H \times W \times 3} \longrightarrow \mathbb{N}^{H \times W \times 1}$$

$$\xrightarrow{f_\theta}$$

1: cow
2: grass
3: tree
4: sky

Cornell Bowers C·IS

Image-to-Image Generation

A *segmentation map* of dimension (H, W, 1) can be viewed as a *generated image*.

Instead of outputting integers for each pixel, the model outputs a vector of length C

$$f_\theta : \mathbb{R}^{H \times W \times 3} \longrightarrow \mathbb{N}^{H \times W \times 1}$$

$$\xrightarrow{f_\theta}$$

1: cow
2: grass
3: tree
4: sky

Image-to-Image Generation

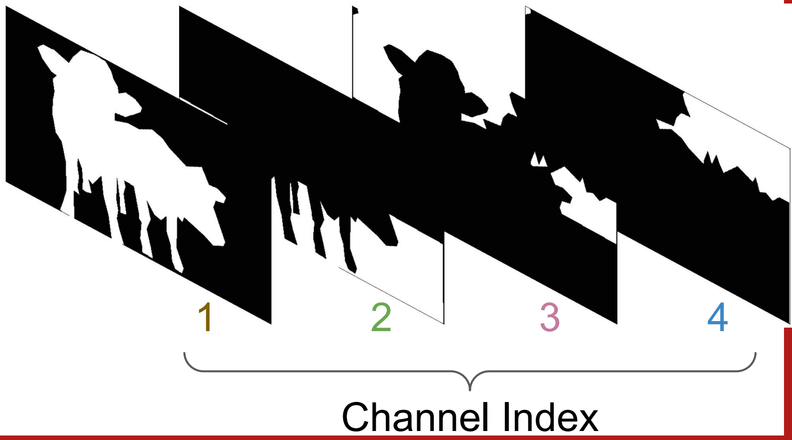
A *segmentation map* of dimension (H, W, 1) can be viewed as a *generated image*.

Instead of outputting integers for each pixel, the model outputs a vector of length C

$$f_\theta : \mathbb{R}^{H \times W \times 3} \longrightarrow \mathbb{R}^{H \times W \times C}$$

$\xrightarrow{f_\theta}$

1: cow
2: grass
3: tree
4: sky

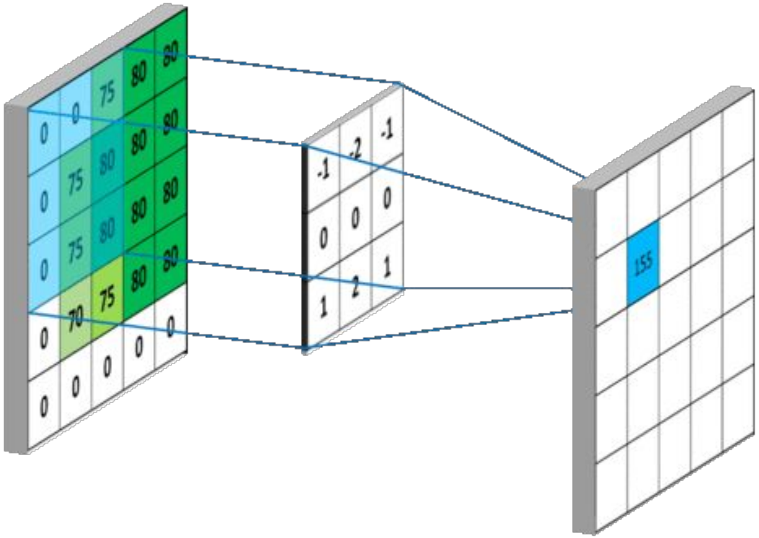


Segmentation Example

Segmentation

Review - Convolutional Neural Network

- ❖ Shared Linear Kernels
- ❖ Translation Invariance
- ❖ Parallel Computation



Building an Image-to-Image Network from Scratch

Input Image

??

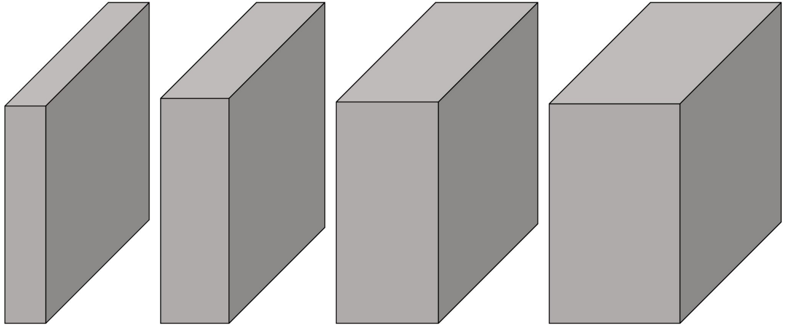
Prediction

Building a Image-to-Image Network from Scratch

✅ Convolutions

Allow parallelization when extracting latent vector for each pixel

Input Image

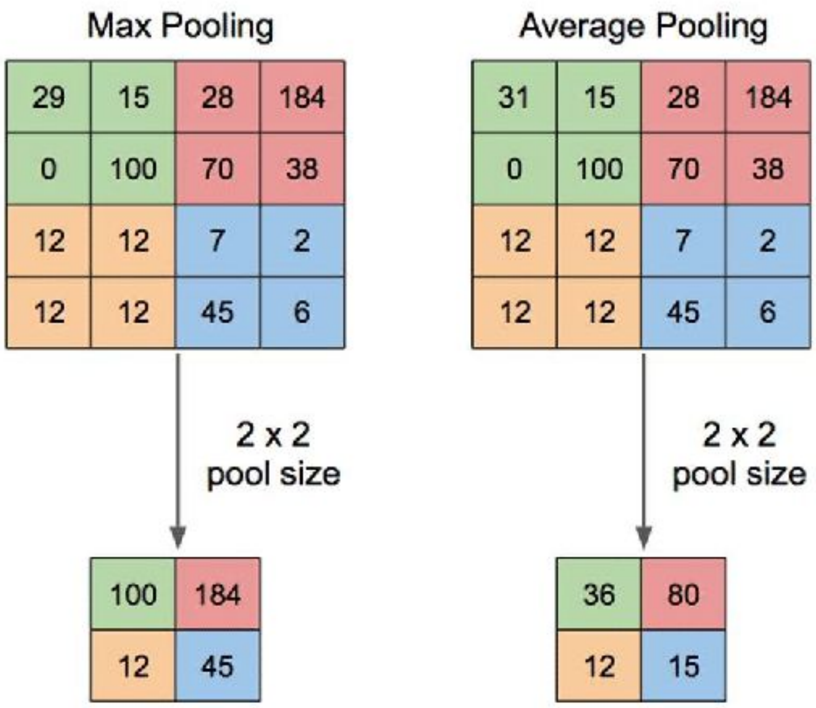


Very Deep CNN at Same Resolution

Prediction

Review - Downsample Pooling

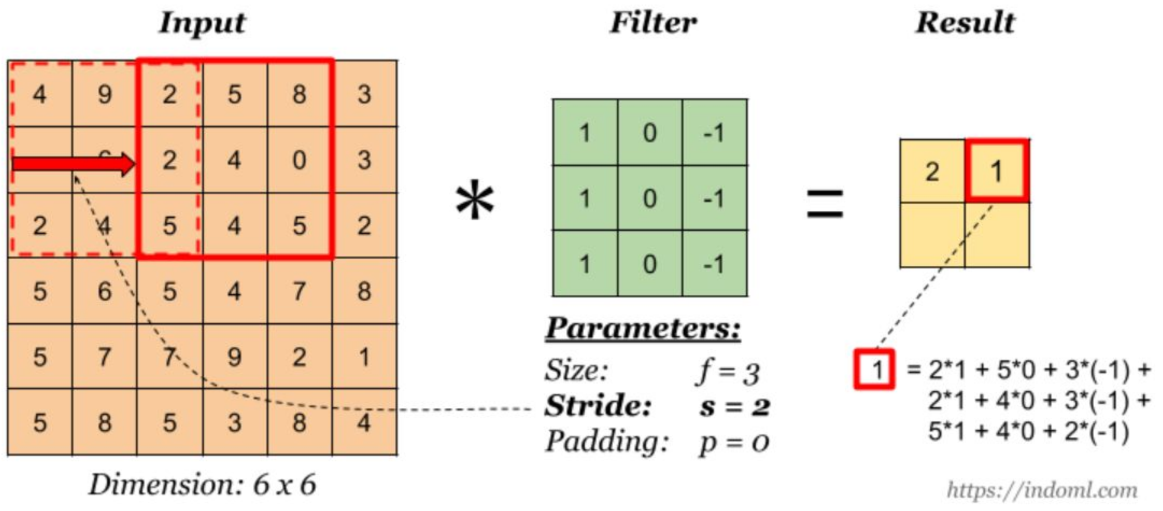
- Down sample feature maps that highlight the most present feature in the patch
- Help over-fitting by providing an abstracted form of representation
- Increase receptive field size



Cornell Bowers C·IS

Review - Strides and Kernel

- ❖ Stride controls how many units the filter / the receptive field shift at a time
- ❖ The size of the output image shrinks more as the stride becomes larger
- ❖ The receptive fields to overlap less as the stride becomes larger



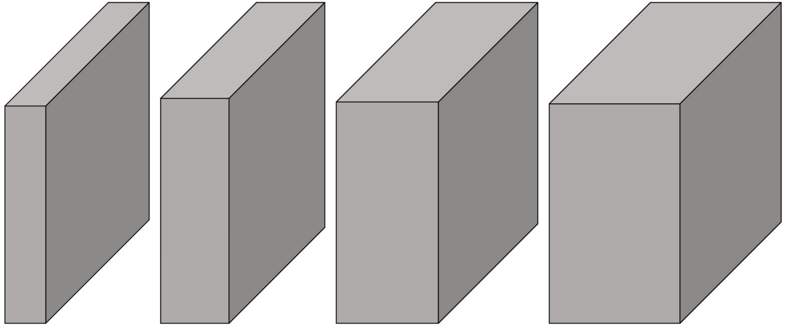
Filter with stride (s) = 2

Cornell Bowers C·IS

Building a Image-to-Image Network from Scratch

✅ **Convolutions** Allow parallelization when extracting latent vector for each pixel

Input Image



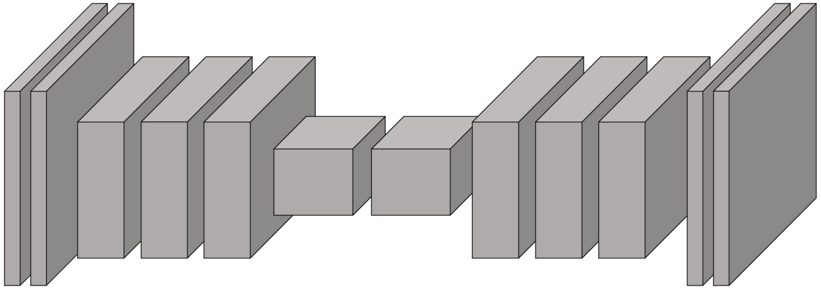
Very Deep CNN at Same Resolution

Prediction

Building a Image-to-Image Network from Scratch

✅ Convolutions	Allow parallelization when extracting latent vector for each pixel
✅ **Hourglass**	Improve efficiency by reducing computations with downsampling
	Increase receptive field size by convolving on downsampled feature maps

Input Image



Hourglass CNN

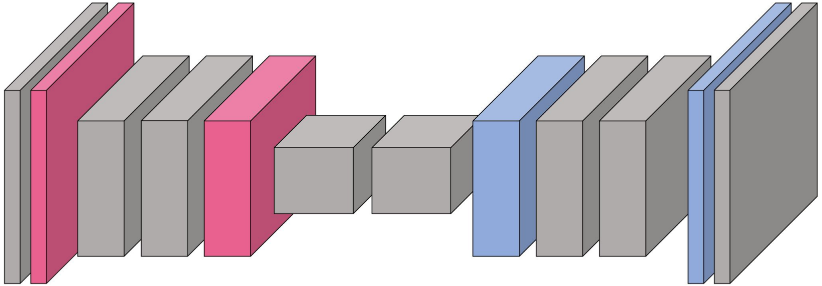
Prediction

Building a Image-to-Image Network from Scratch

(pooling, strided convolution) ← downsample

upsample → (??)

Input Image



Prediction

Cornell Bowers CIS

Upsampling - Unpooling

Nearest Neighbor

1	2
3	4

Input: 2 x 2

1	1	2	2
1	1	2	2
3	3	4	4
3	3	4	4

Output: 4 x 4

"Bed of Nails"

1	2
3	4

Input: 2 x 2

1	0	2	0
0	0	0	0
3	0	4	0
0	0	0	0

Output: 4 x 4

Does not recover all spatial information loss during downsampling!

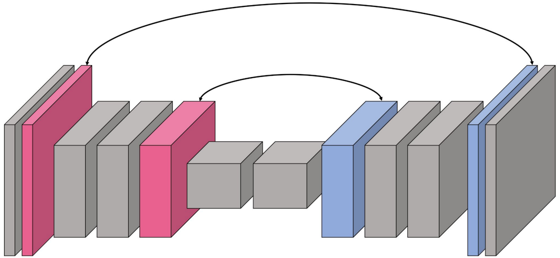
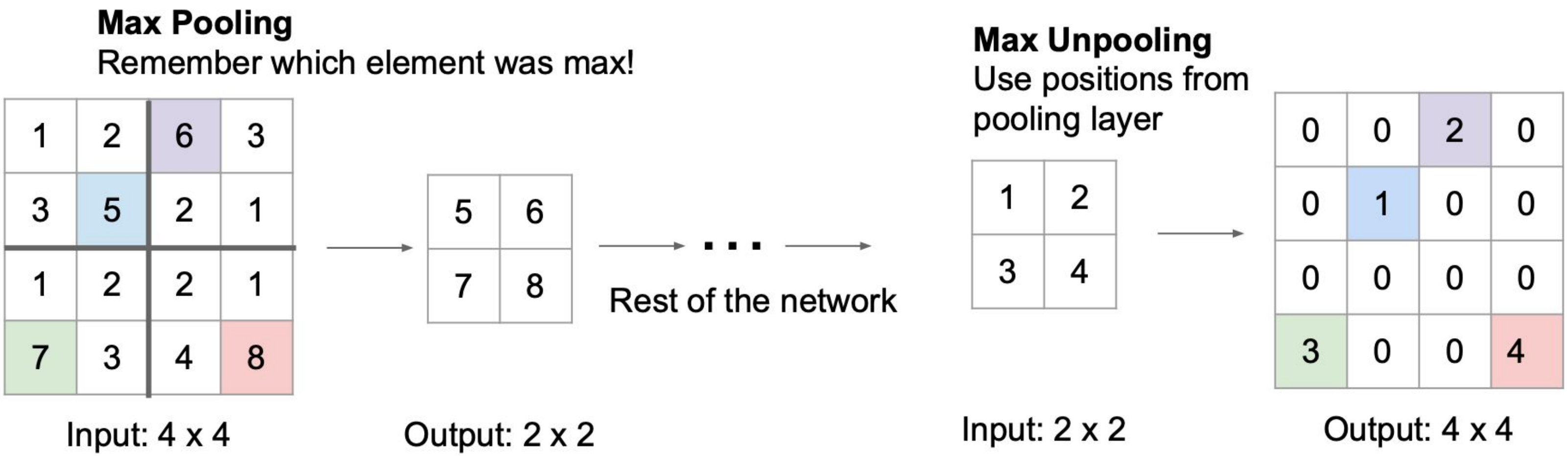


Image Credit: Stanford CS231n, Lecture 11

Upsampling - Max Unpooling



Corresponding pairs of downsampling and upsampling layers

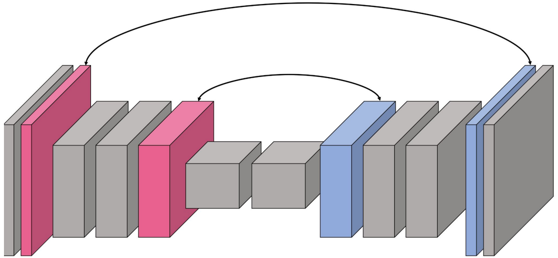


Image Credit: Stanford CS231n, Lecture 11

Cornell Bowers C·IS

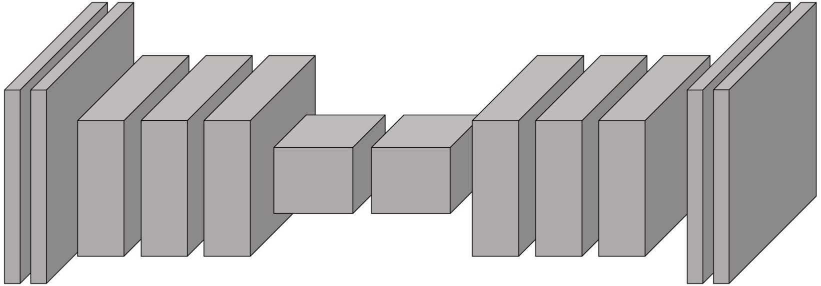
Building a Image-to-Image Network from Scratch

✅ Convolutions — Allow parallelization when extracting latent vector for each pixel

✅ Hourglass — Improve efficiency by reducing computations with downsampling

Increase receptive field size by convolving on downsampled feature maps

Input Image



Hourglass CNN

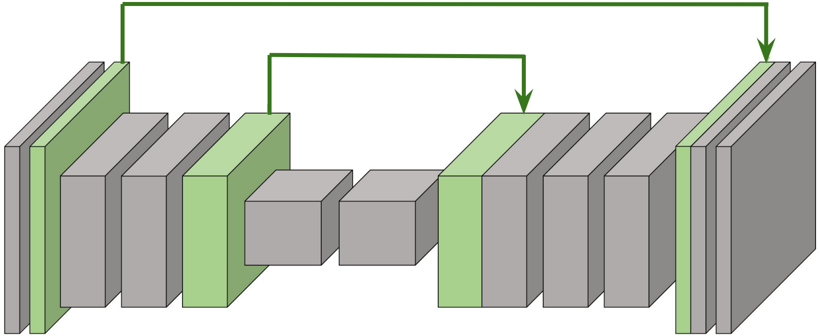
Prediction

Cornell Bowers C·IS

Building a Image-to-Image Network from Scratch

✅ Convolutions	Allow parallelization when extracting latent vector for each pixel
✅ Hourglass	Improve efficiency by reducing computations with downsampling
	Increase receptive field size by convolving on downsampled feature maps
✅ **Skip Connections**	Improve prediction quality by combining low-level image features

Input Image



Hourglass CNN with Skip Connections

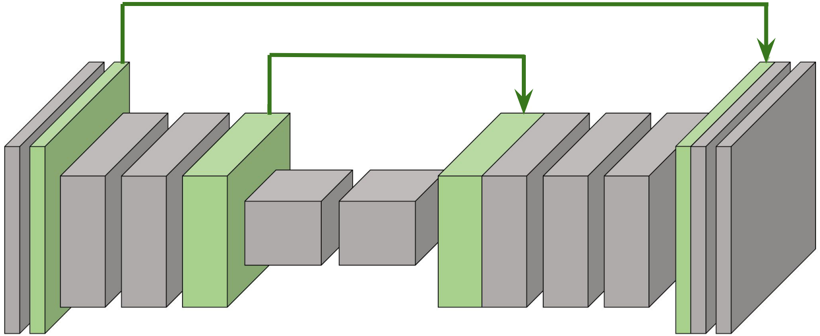
Prediction

U-Net Architecture!

“U-net: Convolutional networks for biomedical image segmentation.” Ronneberger et al., MICCAI 2015

✅ Convolutions	Allow parallelization to extract latent vector for each pixel
✅ Hourglass	Improve efficiency by reducing computations with downsampling
	Increase receptive field size by convolving on downsampled feature maps
✅ Skip Connections	Improve prediction quality by combining low-level image features

Input Image



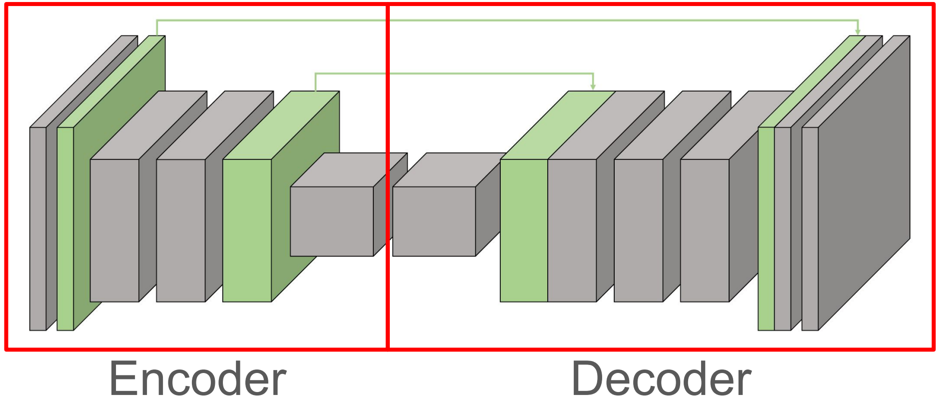
Hourglass CNN with Skip Connections

Prediction

Cornell Bowers CIS

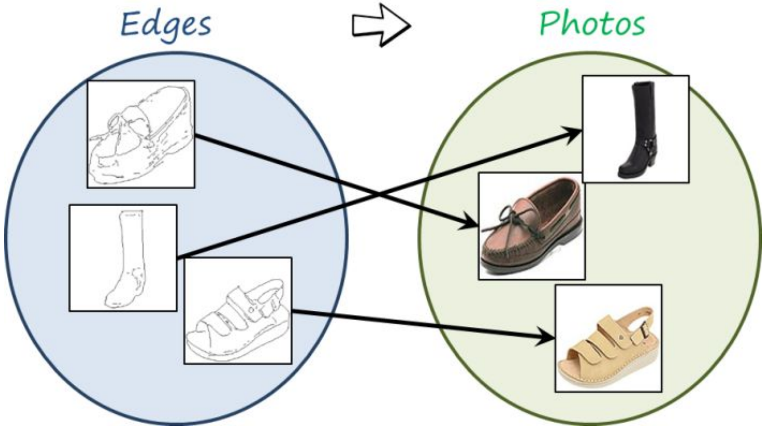
Encoder-Decoder Perspective

- Encoder:
  - Maps an image to a low-resolution, semantically meaningful feature map
  - Basically ResNet!
- Decoder:
  - Maps a low-dimensional feature map to an image
- Can use one or the other depending on the application
  - Similar to transformers!



Unpaired Image Translation

Paired

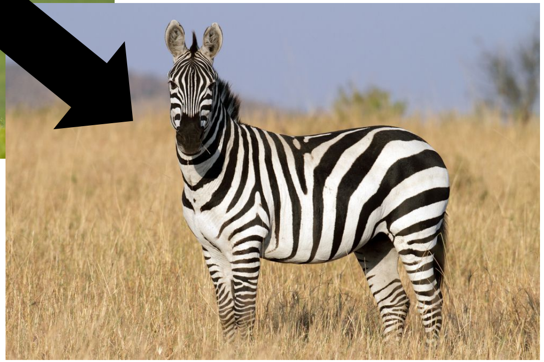


[Generative adversarial networks and image-to-image translation | Luis Herranz]

Paired limitations - it is hard to find exact pairings

Unpaired

[12 Astonishing Facts About Horses]



[Zebra Facts | Live Science]

Cornell Bowers C·IS

Problem with Paired Approaches with Unpaired Translation

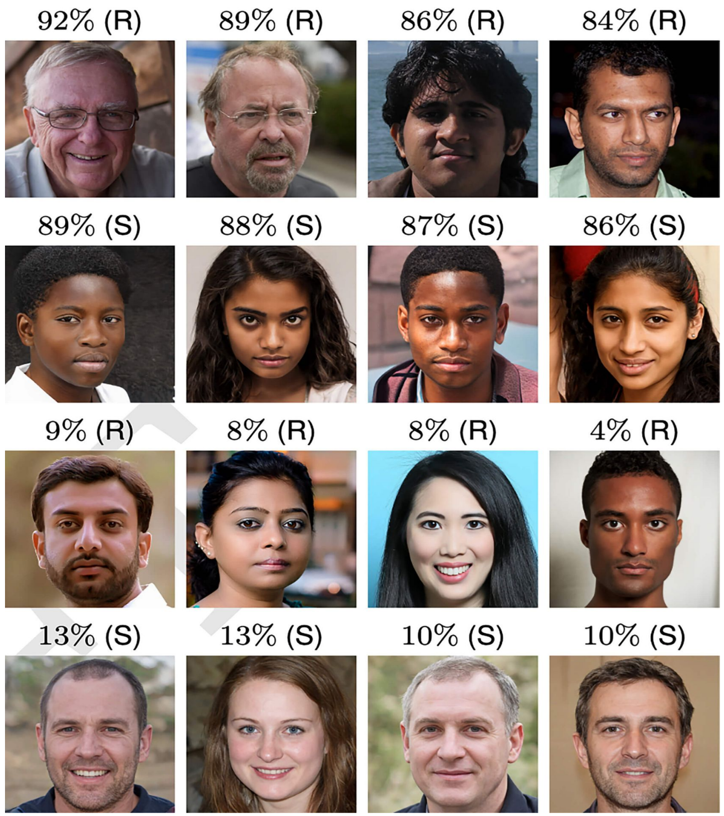
How can we tell if we produced a good output without any reference?

[CycleGAN Project Page]

Cornell Bowers C·IS

Discriminators

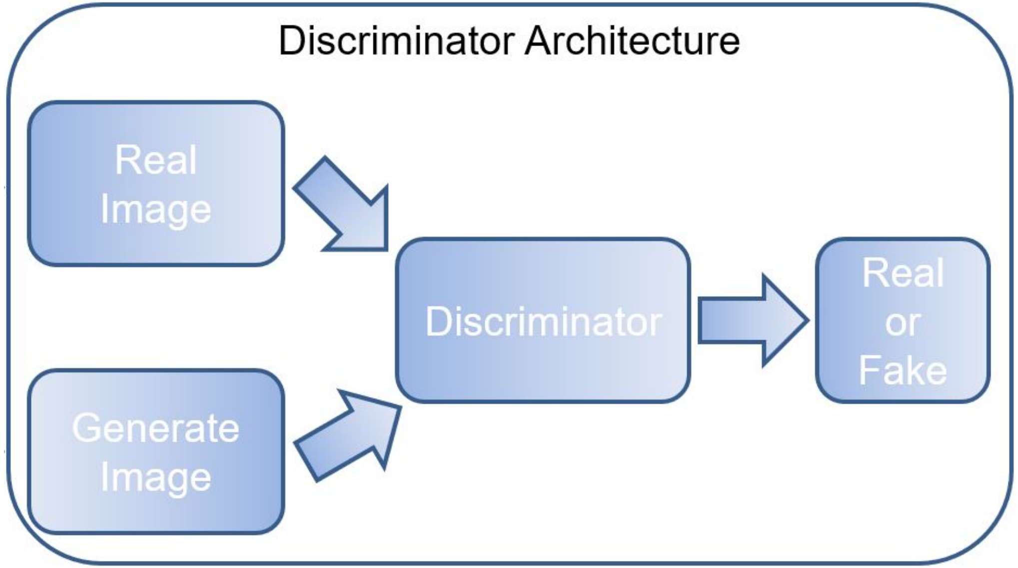
- A binary classifier that determines whether a given image is real or fake
- Can actually use as a learning signal!



https://geneticliteracyproject.org/2022/03/04/can-we-tell-the-difference-between-a-real-human-face-and-an-artificially-generated-one-nope-and-the-ai-ones-are-more-trustworthy/

Discriminator (high-level)

- Supervised machine learning task
- Input pairs features of both real and synthetic data with corresponding labels



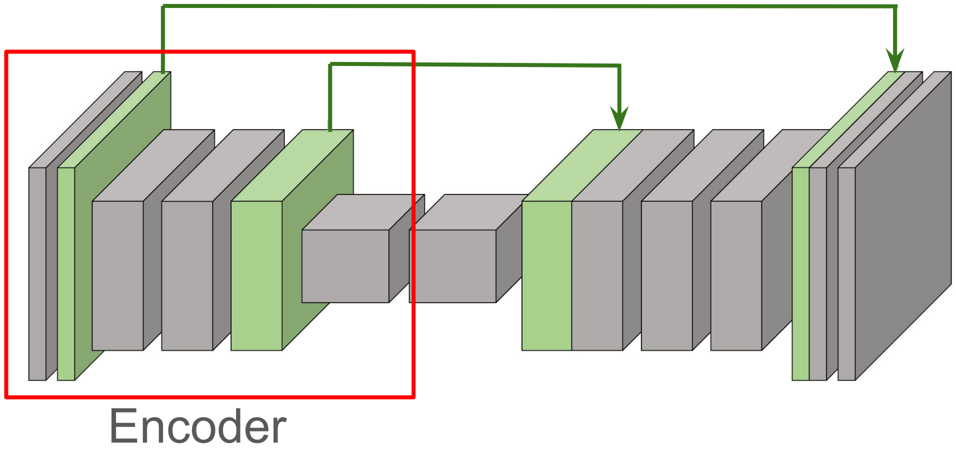
https://subscription.packtpub.com/book/data/9781789139907/1/ch01lvl1sec17/basic-building-block-discriminator

Cornell Bowers C·IS

Discriminator Architecture

- Capture both fine-grained errors and semantic errors
- Fine-grained errors:
  - Blurry/distorted edges
  - Artifacts and Noise
- Semantic errors:
  - Car floating in the air
  - Incorrect textures

A U-Net encoder is a good option for this!



Discriminator as a Classification Model

- Train discriminator with the **binary cross-entropy loss**
- We have the following optimization problem:

$$\min_{D} \left[-y \log(D(x)) - (1-y)\log(1-D(x))\right]$$

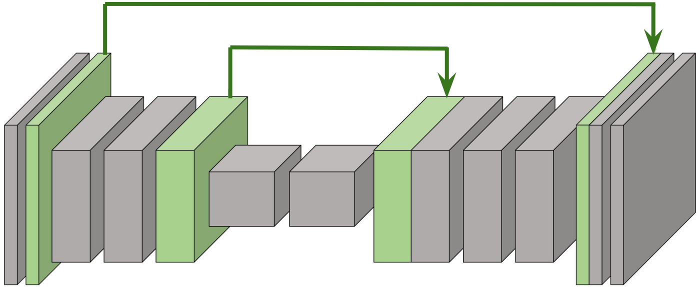
To Generate Realistic Images, We Need:

- A way to generate images (Generator)
- A learning signal to make the images realistic (Discriminator)

Cornell Bowers C·IS

Generator (U-Net)

- Can parameterize our generator as a U-Net!
- How to train?

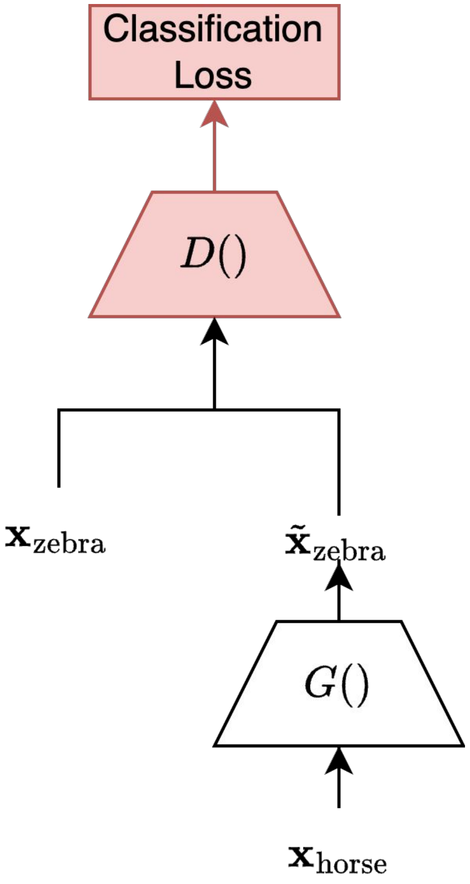


Cornell Bowers C·IS

Discriminator Training

- Train the discriminator to identify real and fake zebra images

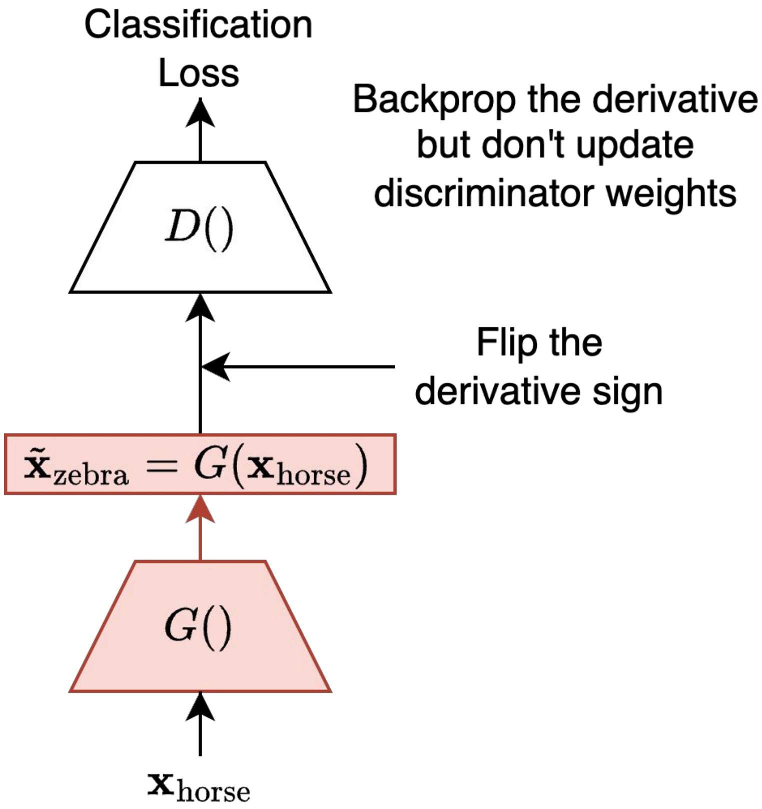
$$\min_{D} \left[-y \log(D(x)) - (1-y)\log(1 - D(x))\right]$$



Cornell Bowers C·IS

Adversarial Training

- Train the generator to fool the discriminator!
    - It should generate images that look like zebras
- Use the negative discriminator loss to update the generator

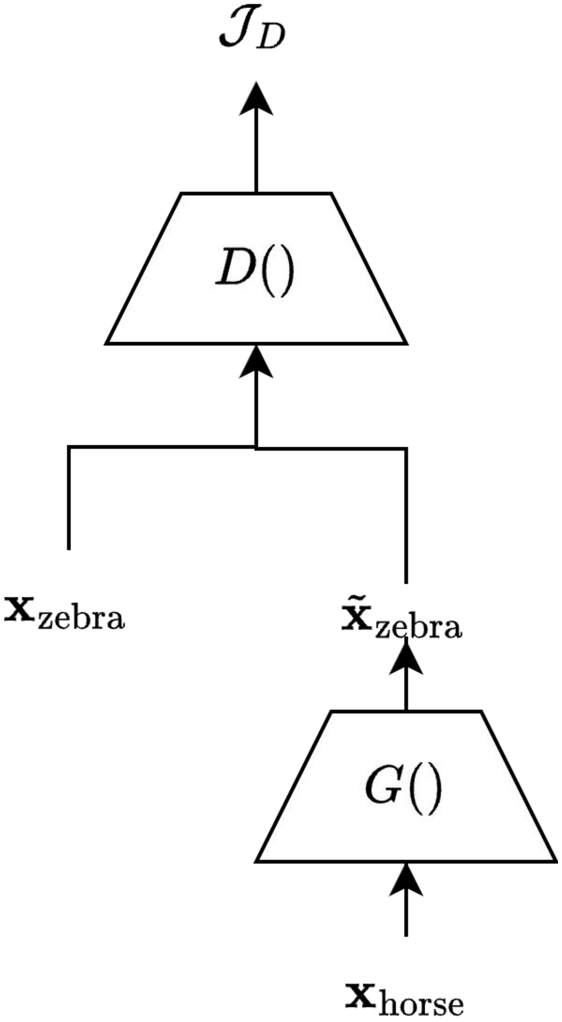


$$\min_G \left[-(1-y)\log(1-D(G(z)))\right] = \min_G \left[-\log(1-D(G(z)))\right]$$

Cornell Bowers C·IS

Generative Adversarial Networks (GANs)

- **Minimax** formulation
  - The generator and discriminator are playing a **zero-sum game** against each other



$$\mathcal{J}_D = \mathbb{E}_{x\sim D}[-\log D(x)] + \mathbb{E}_z[-\log(1 - D(G(z)))]$$

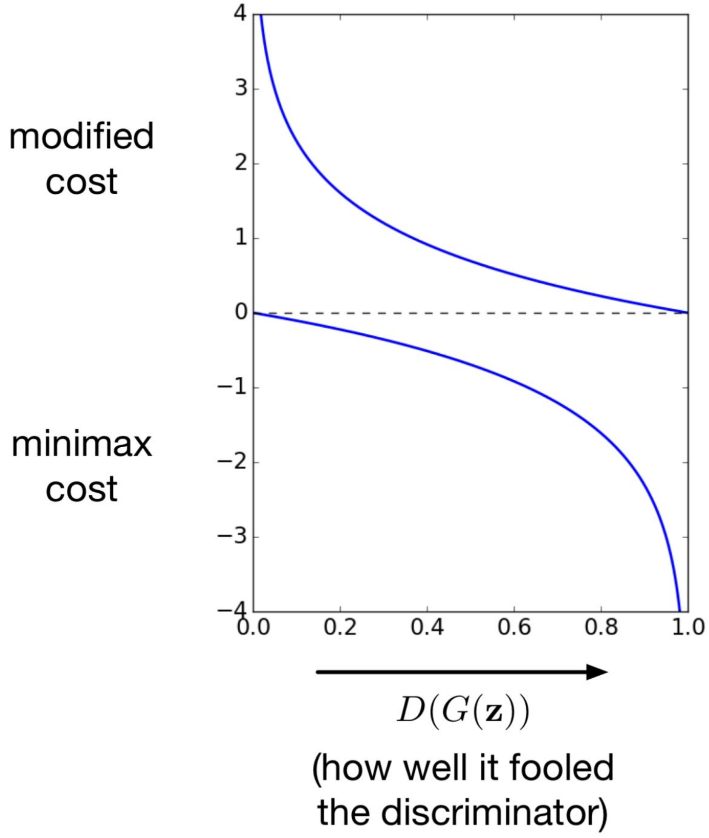
$$\max_G \min_D \mathcal{J}_D$$

Generative Adversarial Networks (GANs)

- Minimax cost runs into vanishing gradient problems with a strong discriminator
  - No learning signal for the generator!

$$\min_G \left[-(1-y)\log(1-D(G(z)))\right] = \min_G \left[-\log(1-D(G(z)))\right]$$

$$\min_G \left[-y\log(D(G(z)))\right] = \min_G \left[-\log(D(G(z)))\right]$$

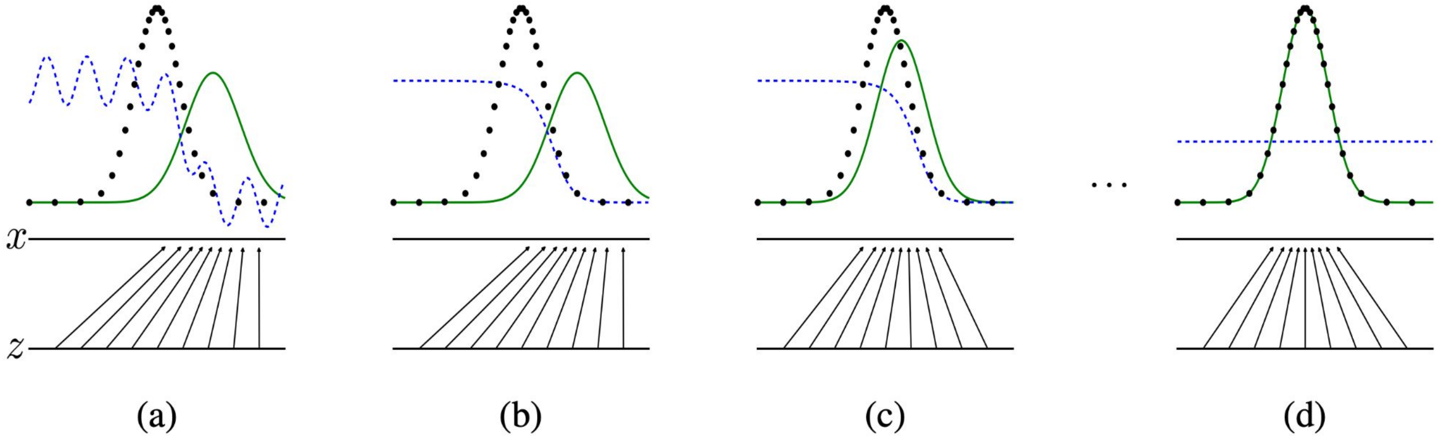


Goodfellow, Ian, et al. "Generative adversarial nets." Advances in neural information processing systems 27 (2014).

Cornell Bowers C·IS

Improving the Generator and Discriminator Together

- The generator and discriminator both start performing poorly
    - Train them together incrementally

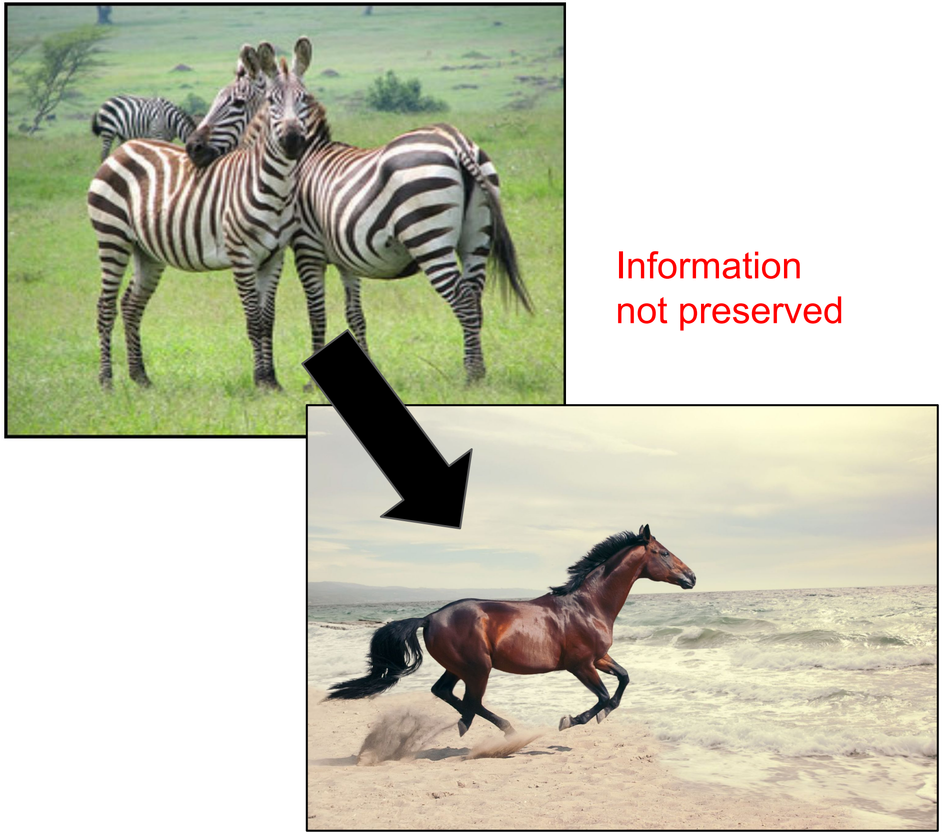


Black dotted line: Data distribution
Green solid line: Generative distribution from Generator (G)
Blue dashed line: Discriminative distribution (D)

Goodfellow, Ian, et al. "Generative adversarial nets." Advances in neural information processing systems 27 (2014).

Unpaired Image Translation

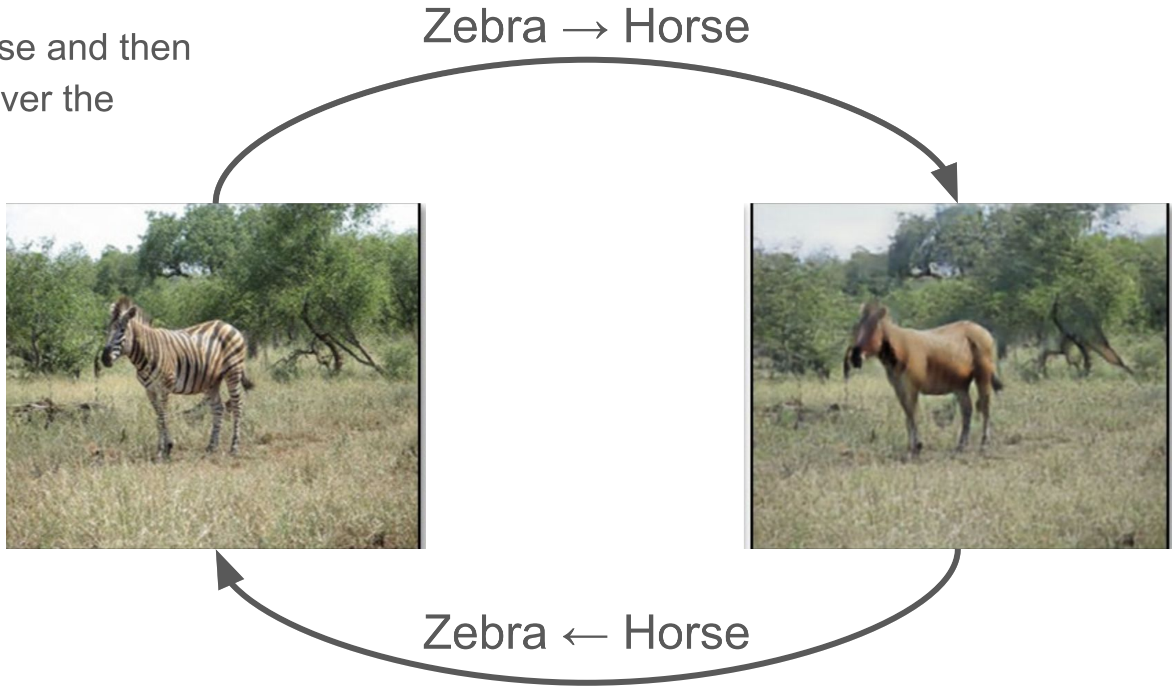
- Want to preserve information about the original image in the generated image



Information not preserved

Key Idea: Cycle Consistency

- Image translation should be **invertible**!
  - Translating a zebra to a horse and then back to a zebra should recover the original image
- Cycle consistency!

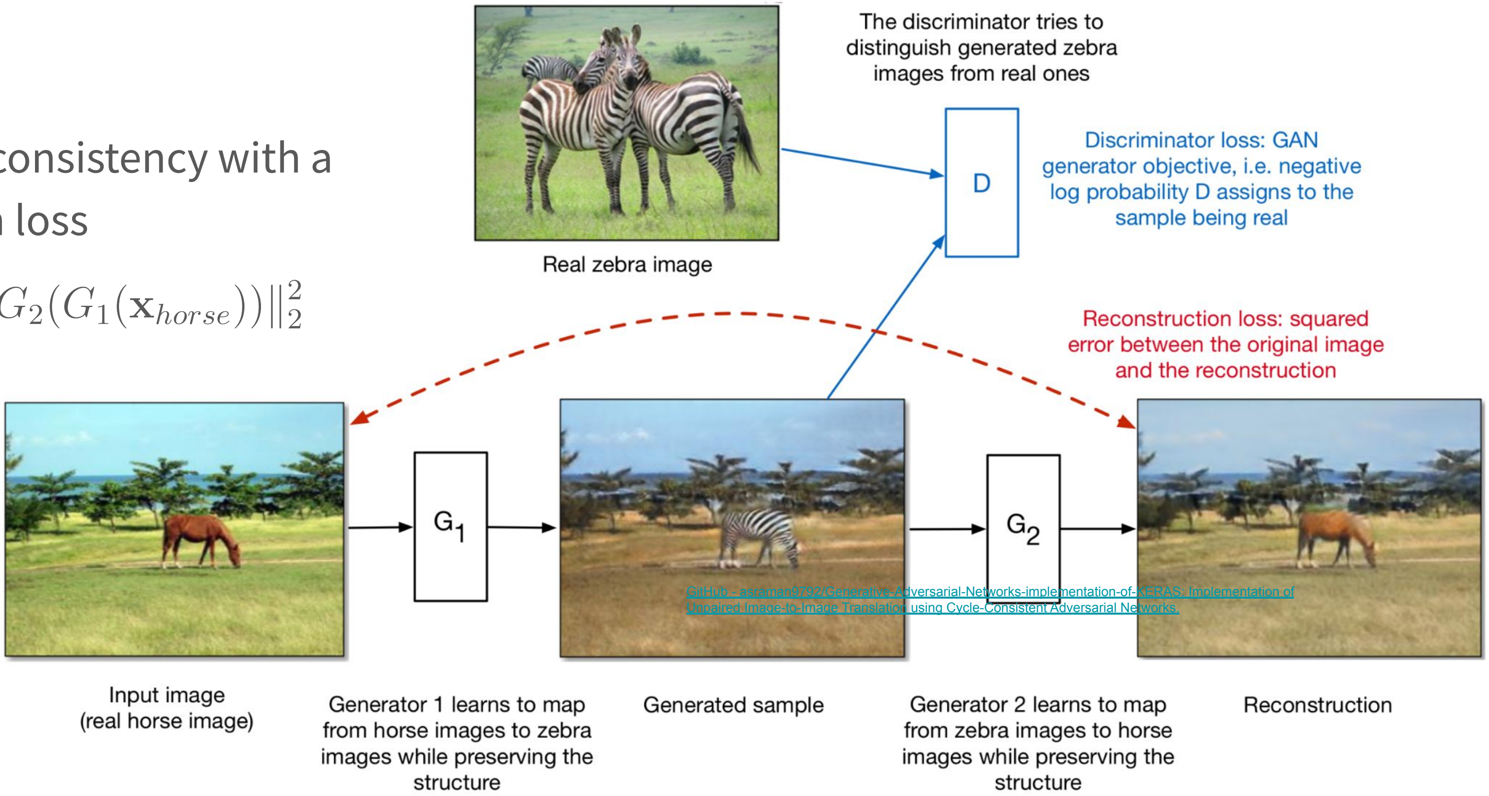


Cornell Bowers C·IS

CycleGAN

- Enforce cycle consistency with a reconstruction loss

$$\|\mathbf{x}_{horse} - G_2(G_1(\mathbf{x}_{horse}))\|_2^2$$



GitHub - asraman9792/Generative-Adversarial-Networks-implementation-of-KERAS: Implementation of Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks.

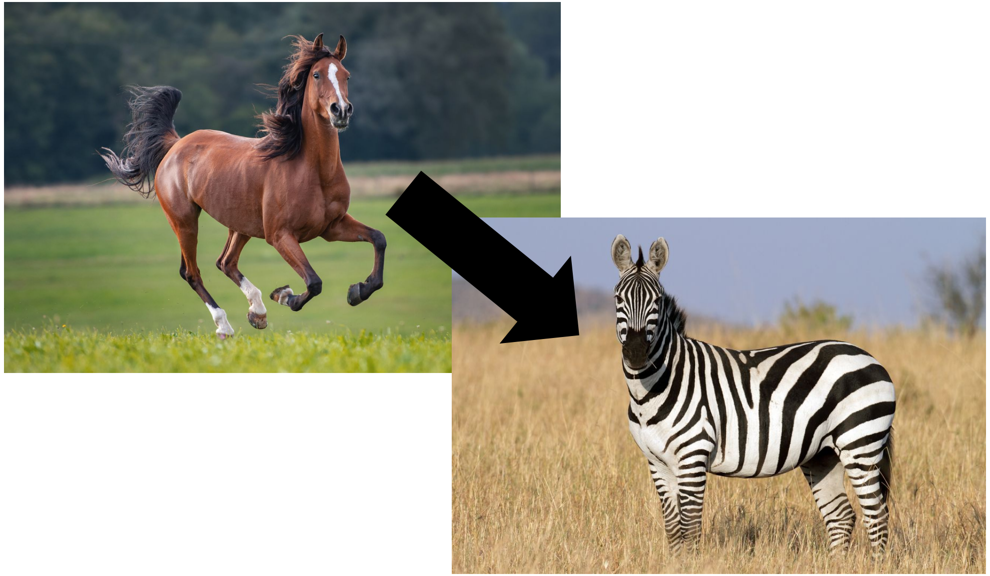
Total loss = discriminator loss + reconstruction loss

Zhu, Jun-Yan, et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks." Proceedings of the IEEE international conference on computer vision. 2017.

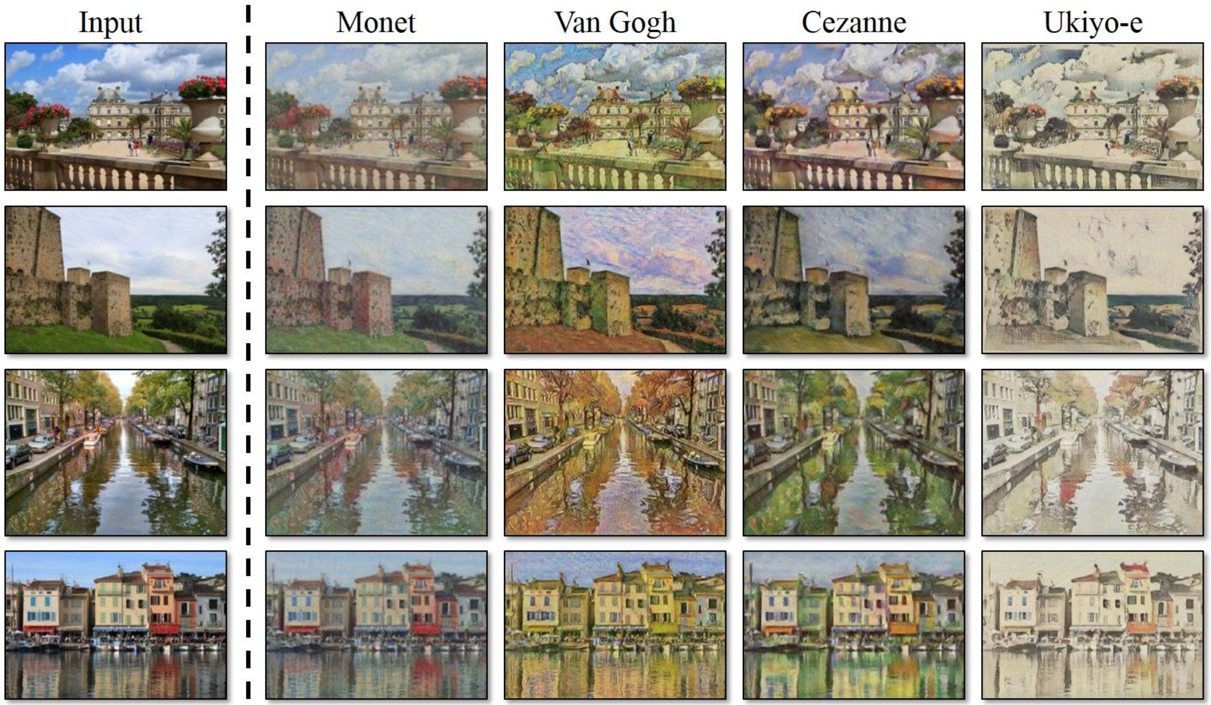
Cornell Bowers C·IS

Discuss: What are some applications of unpaired translation?

Unpaired



The Power of Unpaired Translation



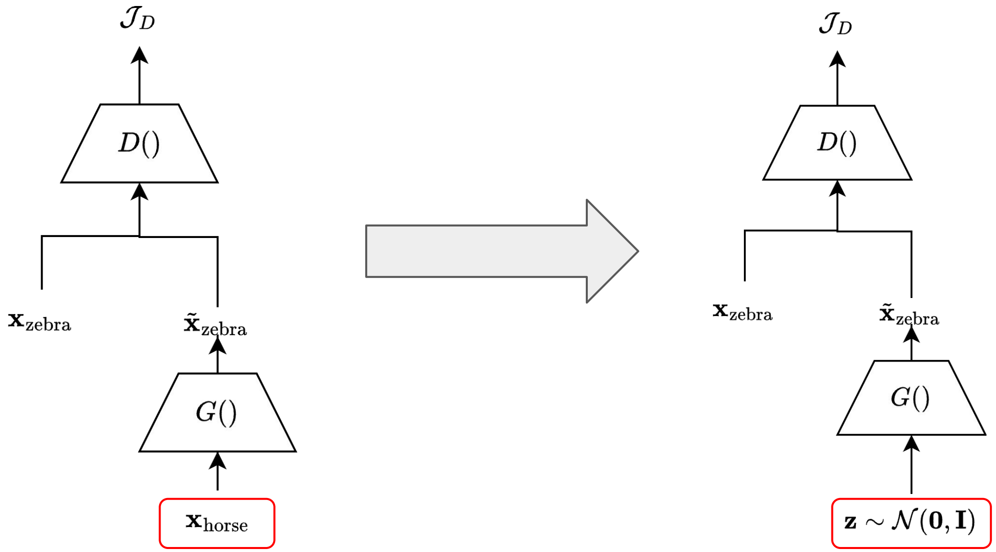
[CycleGAN Project Page]

Zhu, Jun-Yan, et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks." Proceedings of the IEEE international conference on computer vision. 2017.

Cornell Bowers C·IS

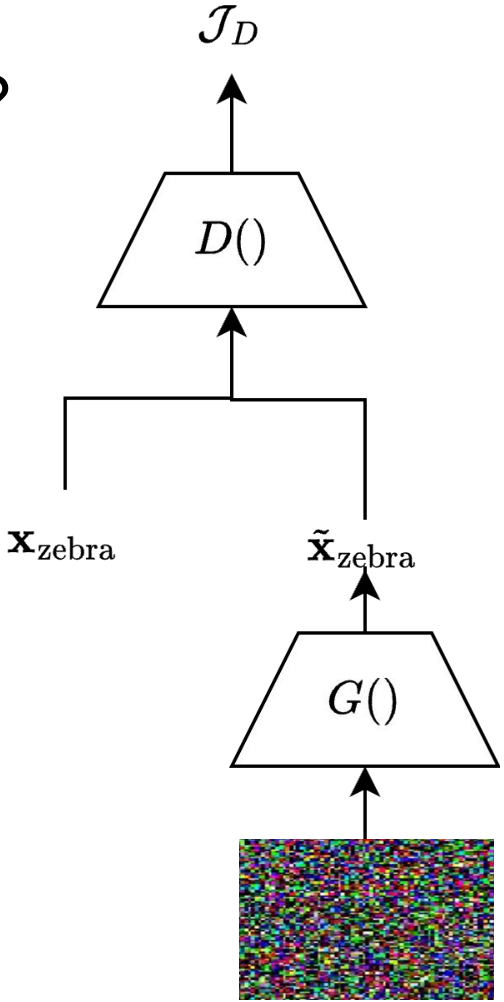
Can we perform unconditional generation?

- Just want to draw samples from some distribution of images (e.g. zebras)
- Replace the source image with Gaussian noise



Can we perform unconditional generation?

- Just want to draw samples from some distribution of images (e.g. zebras)
- Replace the source image with Gaussian noise



Cornell Bowers C·IS

GANs

- First generative model capable of realistic high-resolution image synthesis
- Very fast generation!

Latent Space Properties

Gaussian noise vector is meaningful

- Similar noise vectors lead similar images
- Noise dimensions are meaningful!
- Can exploit this to control generation

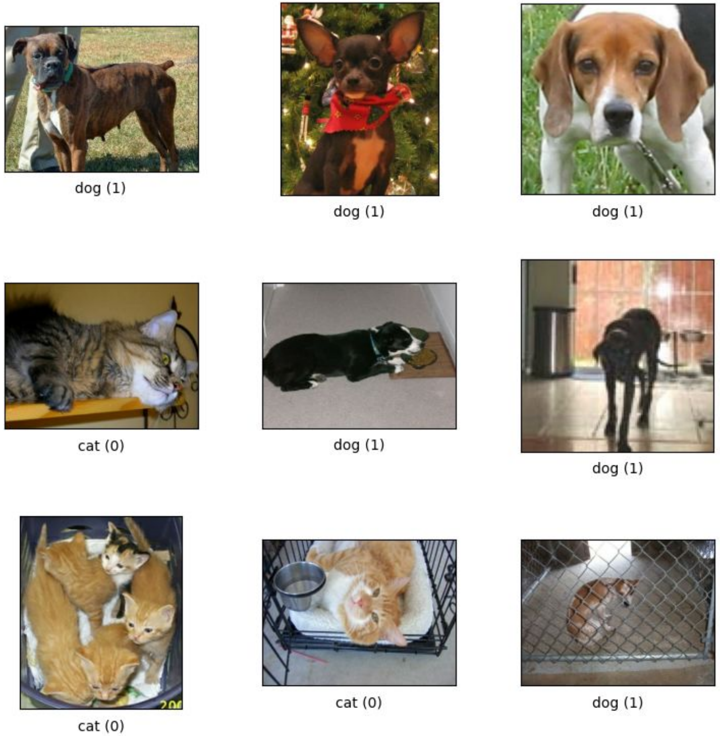
Cornell Bowers C·IS

Limitations of GANs

- Minimax training objective is hard to optimize!
  - Can lead to oscillations/instability during training
- Not guaranteed to converge to a good solution
  - Sensitive to hyperparameter settings, network architectures, and the choice of loss functions

Limitations of GANs

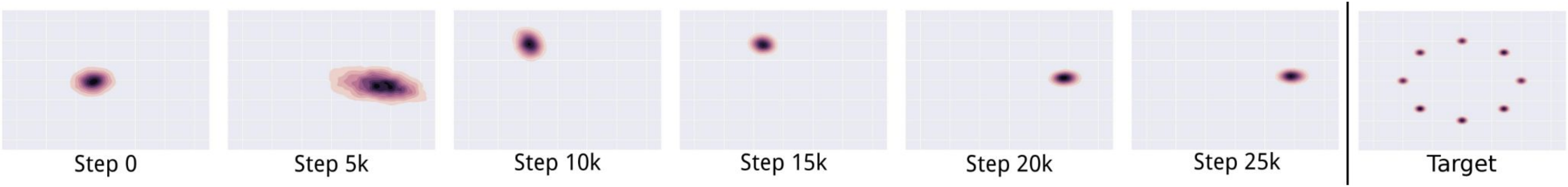
- Consider training a GAN on a dataset of dogs and cats
- Generator could specialize in generating realistic dogs
  - Successfully fools the discriminator!



Cornell Bowers C·IS

Mode Collapse

- Big problem in practice
- GANs often fail to model the full distribution of images
    - “Collapse” to some popular mode to fool the discriminator



Metz, Luke, et al. "Unrolled generative adversarial networks." arXiv preprint arXiv:1611.02163 (2016).

Recap

- Many vision tasks can be formulated as image-to-image problems
  - Segmentation, super-resolution, etc.
- The U-Net is a versatile encoder-decoder architecture for these tasks
- CycleGAN can perform unpaired image translation by learning to fool a discriminator
- GANs can perform unconditional image generation conditioned on samples of Gaussian noise
  - Challenging to train
  - Susceptible to mode collapse