CS181: Computer Vision

Generative Modeling
Generative Adversarial Network (GAN)

October, 29th, 2026



Md Alimoor Reza
Associate Professor of Computer Science

Can you recognize any famous person here?



A style-based generator architecture for generative adversarial networks - Tero Karras, Samuli Laine, Timo Aila - arXiv18

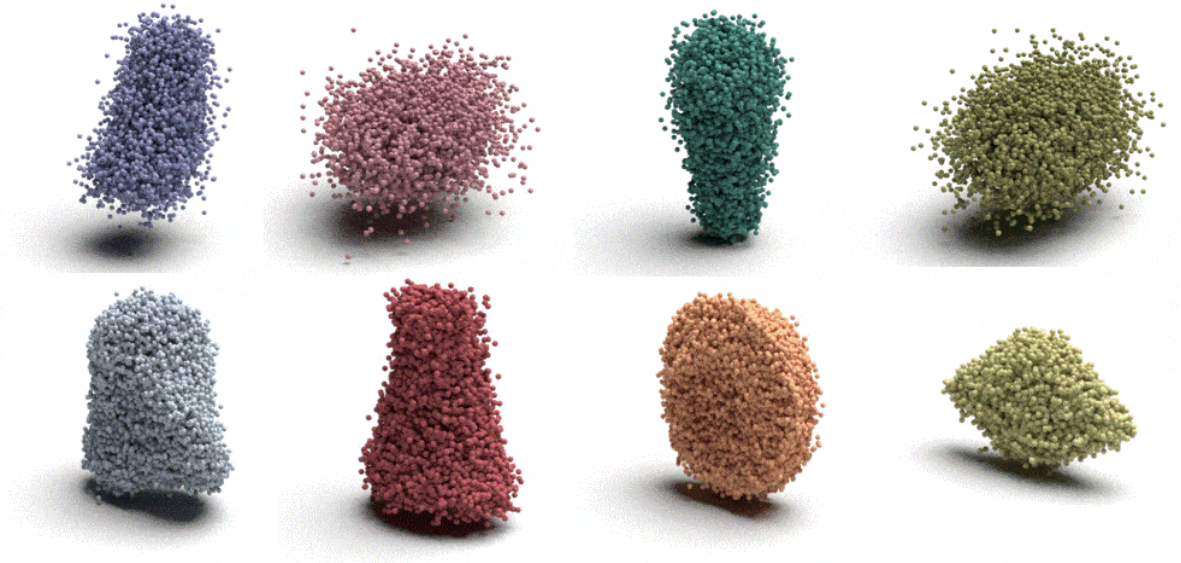
Today's Topics

- **Generative Modeling**
- Introduction to generative adversarial network (GAN)
- GAN variations
- Computer vision applications with GAN
- GAN model in my research

Generative Model

- Generative modeling have achieved state-of-the-art performance on many downstream tasks and applications

- Shape generation: generate shapes from 3D point-cloud



Learning Gradient Fields for Shape Generation - ECCV'20

Generative Model

- Generative modeling can be used on many downstream tasks and applications

- Audio synthesis: generate audio that sounds realistic

DiffWave: A Versatile Diffusion Model for Audio Synthesis - ICML'21

- Music generation: generate new music

Symbolic Music Generation with Diffusion Models

Generative Model: Latent Space

- A dataset typically exists in a high-dimensional space, but it is often possible to identify a lower-dimensional subspace that captures the essential variation among the samples.

- **Big Idea**: Find new (or *latent* features) made up of combinations of existing features.
  - Maybe multiplying `sepal length*petal width` is more helpful in identifying the species of an iris than either `sepal length` or `petal width` are on their own

Generative Model: Latent Space

- *measurable feature (original representation)* vs. *latent features*

- Imagine we are attempting to predict the price of a house based on the following measurable features:
  - *house square footage*
  - *number of rooms*
  - *school district test scores*
  - *neighborhood amenities eg, park and recreation, grocery shops, etc*

Generative Model: Latent Space

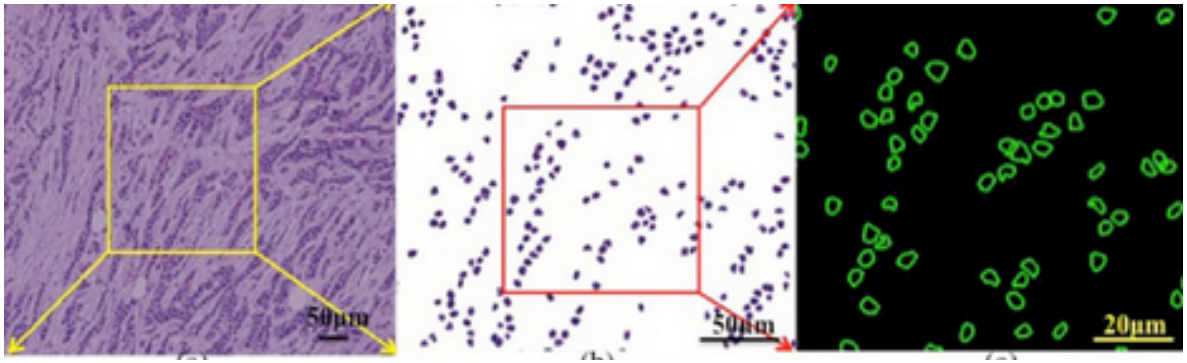
- **Claim**: In the 4 *measurable features*, there are really only **2 latent features** which explain these four measurable features. In other words, there are two composite features that more directly probe the underlying phenomenon of the data
    - *house square footage*
    - *number of rooms*
    - *school district test scores*
    - *neighborhood amenities eg, park and recreation, grocery shops, etc*
- Can you see the pattern and guess what these **two latent features** are?

Generative Model: Latent Space

- **Claim**: In the 4 *measurable features*, there are really only **2 latent features** which explain these four measurable features. In other words, there are two composite features that more directly probe the underlying phenomenon of the data
    - *house square footage*
    - *number of rooms*
    - *school district test scores*
    - *neighborhood amenities eg, park and recreation, grocery shops, etc*
- Can you see the pattern and guess what these **two latent features** are?
    - Size of house:
        - *house square footage*
        - *number of rooms*
    - Location of house:
        - *school district test scores*
        - *neighborhood amenities eg, park and recreation, grocery shops, etc*

Generative Model: Latent Space

- measurable feature vs. latent features



- Imagine we are attempting to predict a breast-cancer histopathology image based on the following latent features:

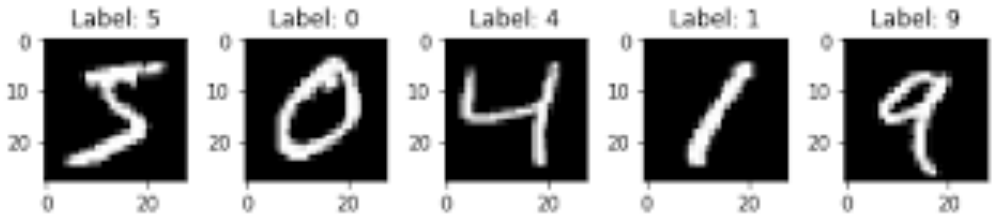
Shape-based features
- Area
- Perimeter
- Circularity
- Aspect ratio
- Distance transform
- Line sweep

Architectural features
- Distance transform (global)
- Line sweep (global)

Mark Zarella, David Breen, Alimoor Reza, Aladin Milutinovic, and Fernando Garcia. Lymph Node Metastasis Status in Breast Carcinoma Can Be Predicted via Image Analysis of Tumor Histology - Journal of Analytical Quantiative Cytopathology and Histopathology'2015

Generative Model: Latent Space

- Let's consider the MNIST dataset, which is a collection of 28x28 pixel images. Each image represents one of 10 possible digits

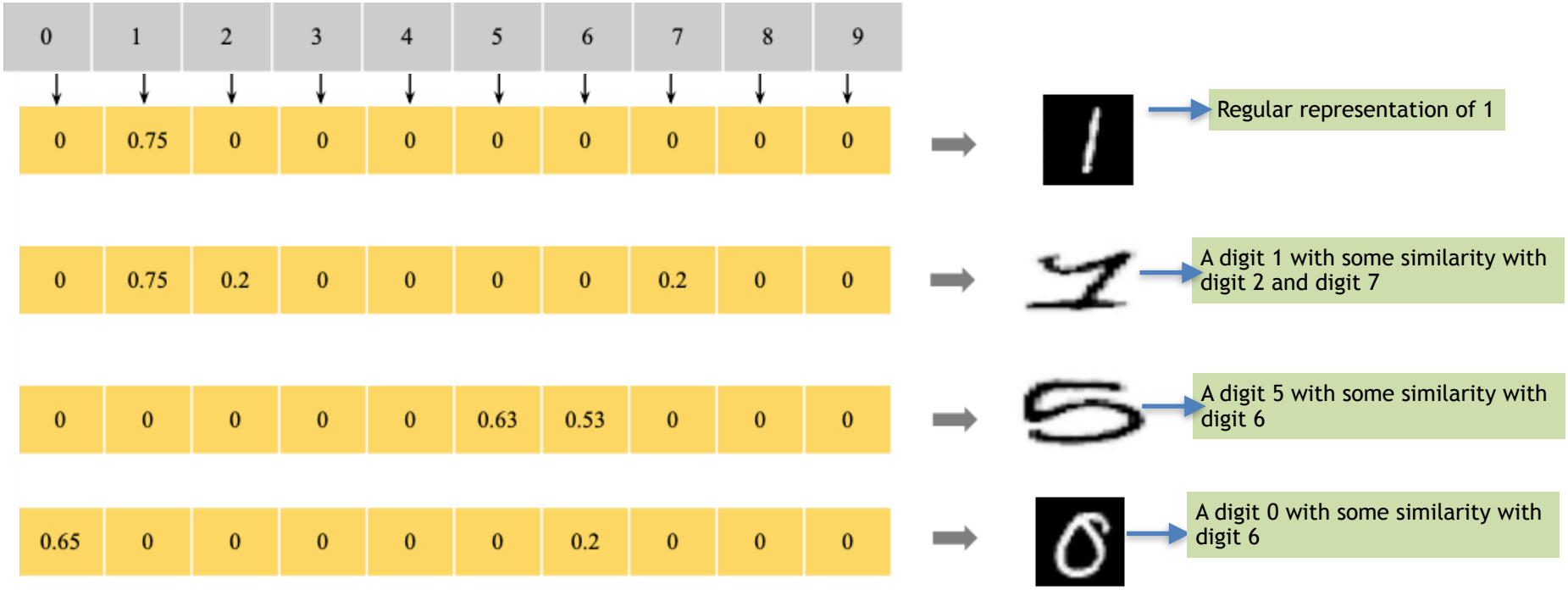


- Images are represented in a 28*28 = 768 dimensional pixel space
  - Most of the pixels are not useful (eg, a lot of black region in each image with no information)
  - Hence, most of the dimensions are redundant

- Such images can be represented in a much smaller dimensions
  - significantly less than 768 dimensional pixel space
  - Can you think of any 10 dimensional or 15 dimensional feature space?

10 Dimensional Latent Space

- 10 dimensional latent representation:
  - 1 dimension to represent each digit
  - It could contain a floating point number signifying the variations



- This 10 dimensional latent representation is known as latent vector (LV)
- The size of the latent vector is 10, which is a hyper-parameter
- How can you generate this?

Generative Model

- Generative modeling is based on representation of probability distribution
- Neural networks are universal function approximators, hence they are also good at approximating any desired probability distribution with high accuracy
- Generative models use machine learning algorithms to learn a probability distribution from a set of training data and then generate new examples from that distribution
  - Think of such a generative model in terms of a distribution $p(x|w)$ in which x is a vector in the data space, and w represent the learnable parameters of the neural network model
  - In many cases we are interested in conditional generative models of the form $p(x|c, w)$ where c represents a vector of conditioning variables
  - For example, a generative model trained on animal images and it's used to generate new image of an animal. We could specify by variable c which animal image we want to generate

Types of Generative Models

- Existing generative modeling techniques can be classified into four broad categories
  - Generative adversarial networks GAN (implicit generative models)
  - Variational auto-encoders VAE (likelihood-based models)
  - Diffusion models (score-based models)
  - Normalizing flows

Generative Model: Generative adversarial networks (GAN)

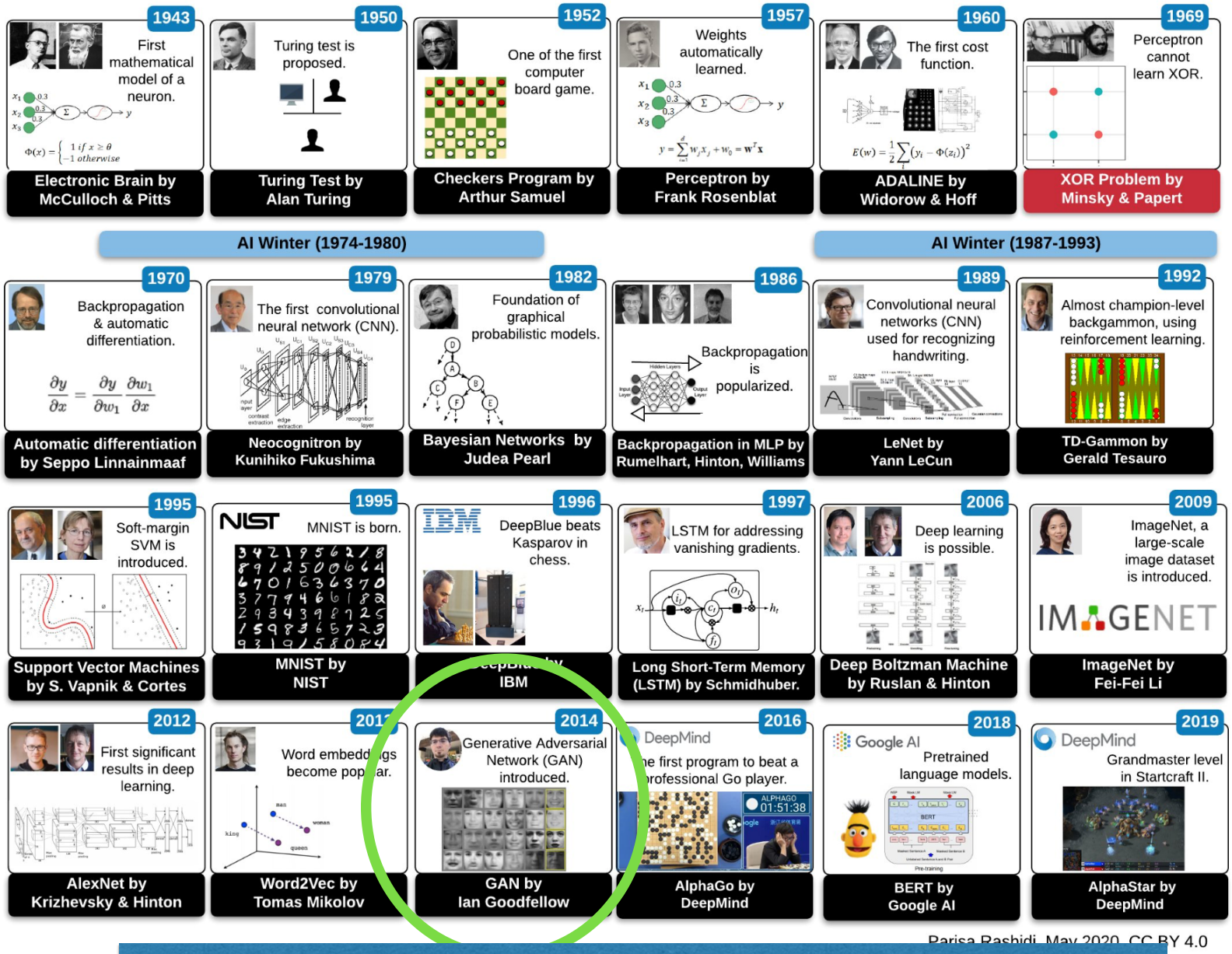
- **Generative Adversarial Network (GAN):**
  - GAN is an example of implicit generative model. It implicitly represents a distribution over all objects that can be produced by the *generator network which is trained from the data*
  - probability distribution is represented by a model of its sampling process

  - Limitations
    - requires adversarial training which is very unstable (eg, due to mode collapse)

Today's Topics

- **Generative Modeling**
- **Introduction to generative adversarial network (GAN)**
- GAN variations
- Computer vision applications with GAN
- GAN model in my research

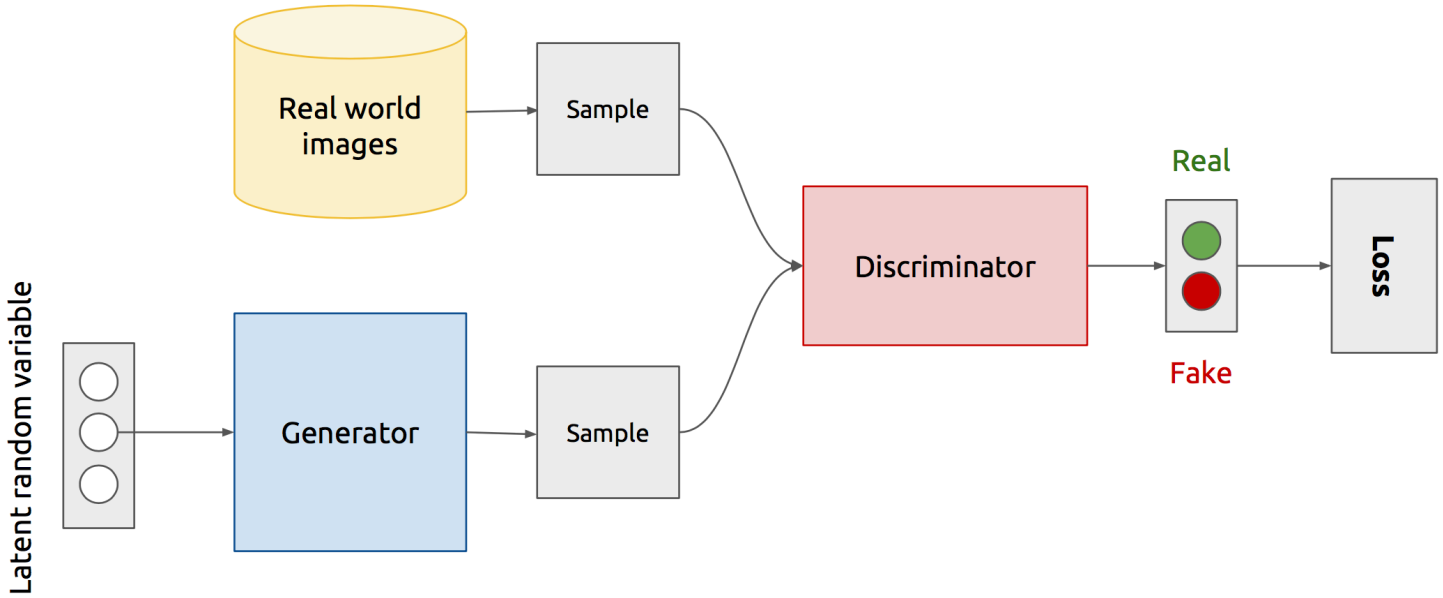
Where is GAN in AI's historical milestones?



Parisa Rashidi, May 2020, CC BY 4.0

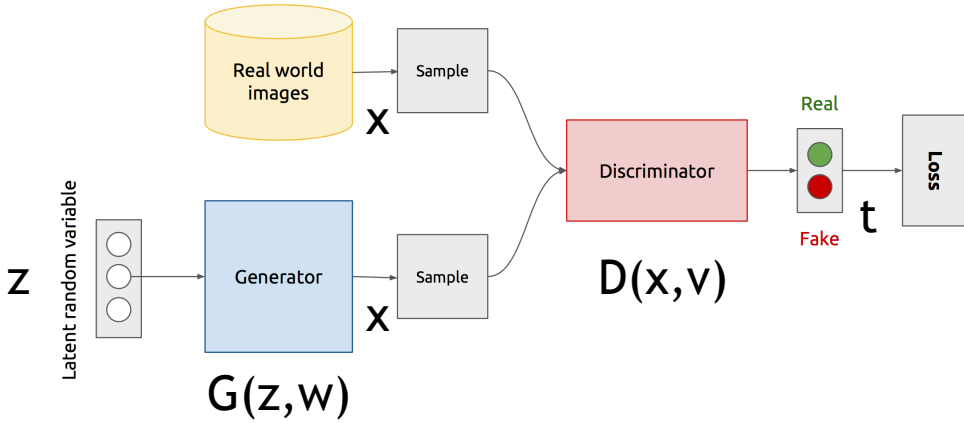
What is GAN — intuitive introduction?

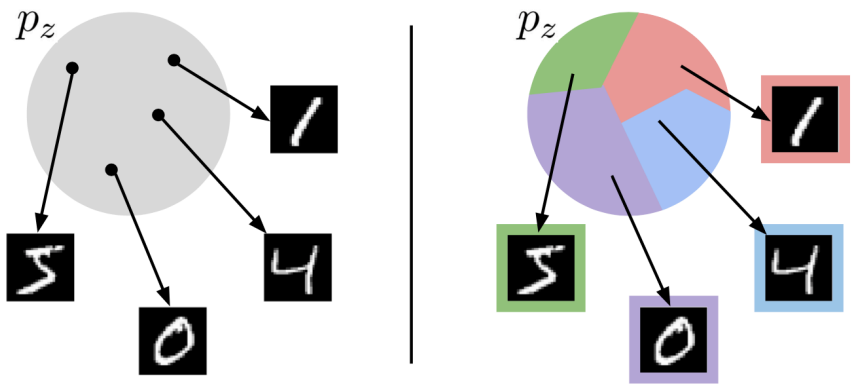
- Adversarial play between a two entities: a generator and a discriminator
  - Generator
  - Discriminator



Random variable as input to Generator

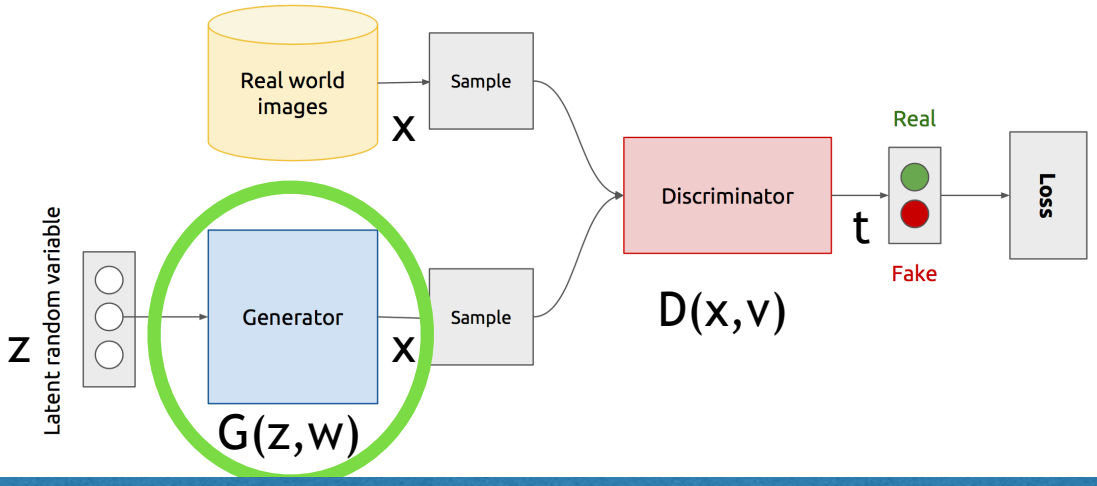
- Latent random variable: something that's easy to sample from, like a uniform or Gaussian distribution
- Generator then transforms this noise into a meaningful output
- GAN tries produce a wide variety of data, sampling from different places in the target distribution





Generator training

- Sample random noise
- Produce generator output from sampled random noise
- Let discriminator determine "Real" (label=1) or "Fake" (label=0) classification for generator output
- Calculate loss from discriminator classification
- Back-propagate through both the discriminator and generator to obtain gradients
- Use gradients to change only the generator weights

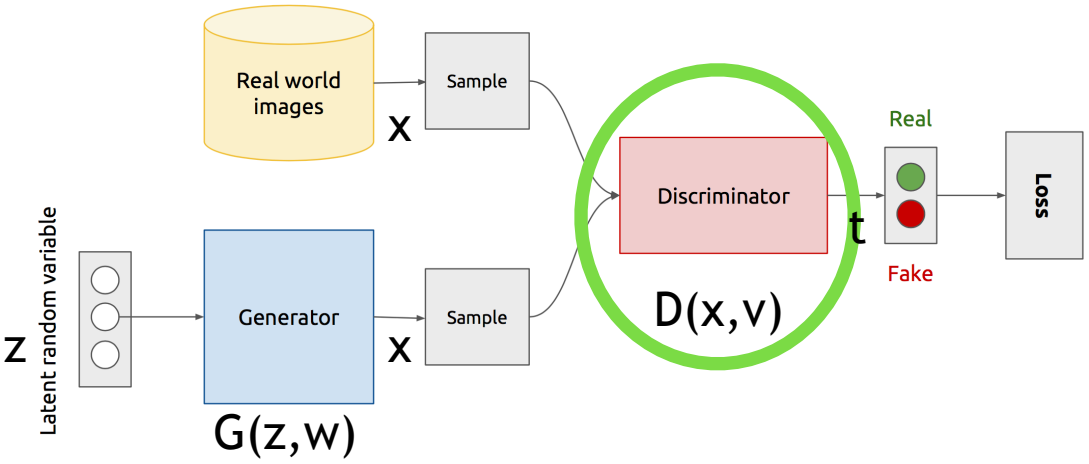


Discriminator training

- The discriminator classifies both "Real" (label=1) and "Fake" (label=0) from the generator
  - Two sets of inputs: ”Real" (label=1) and "Fake" (label=0)

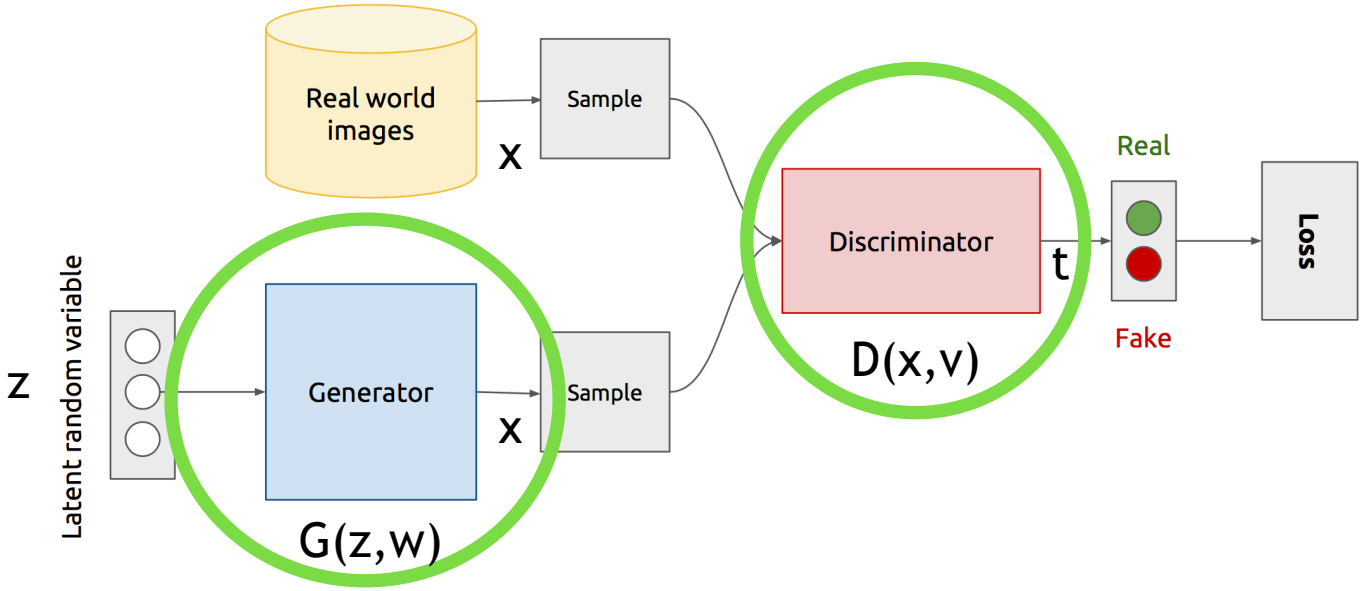
- The discriminator loss penalizes the discriminator for misclassifying a real instance as fake or a fake instance as real

- The discriminator updates its weights through backpropatation from the discriminator loss



Overall GAN training

- GAN training proceeds in alternating periods (for several times until convergence):
  - The discriminator trains for one or more epochs
  - The generator trains for one or more epochs



How does GAN work — mathematical model?

- Generator tries to minimize the loss function while the discriminator tries to maximize

$$E_x[log(D(x))] + E_z[log(1 - D(G(z)))]$$

- Discriminator tries gradient ascent to maximize using the following cost

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

- Generator tries gradient descent to minimize using the following cost

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right)$$

Today's Topics

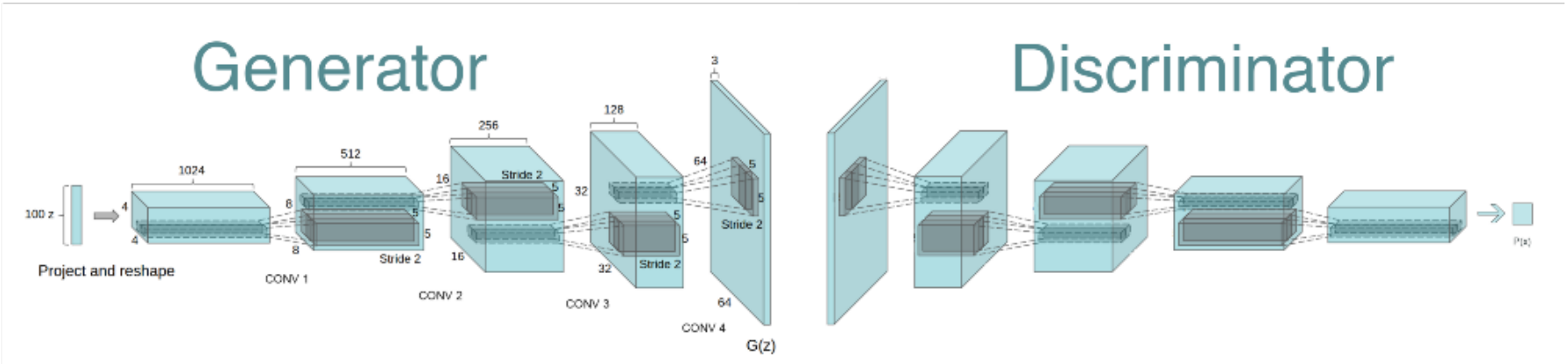
- **Generative Modeling**
- Introduction to generative adversarial network (GAN)
- **GAN variations**
- Computer vision applications with GAN
- GAN model in my research

GAN variations

- Deep Convolutional GAN (DCGAN)
- Conditional GAN (cGAN)
- Wasserstein GAN (WGAN)

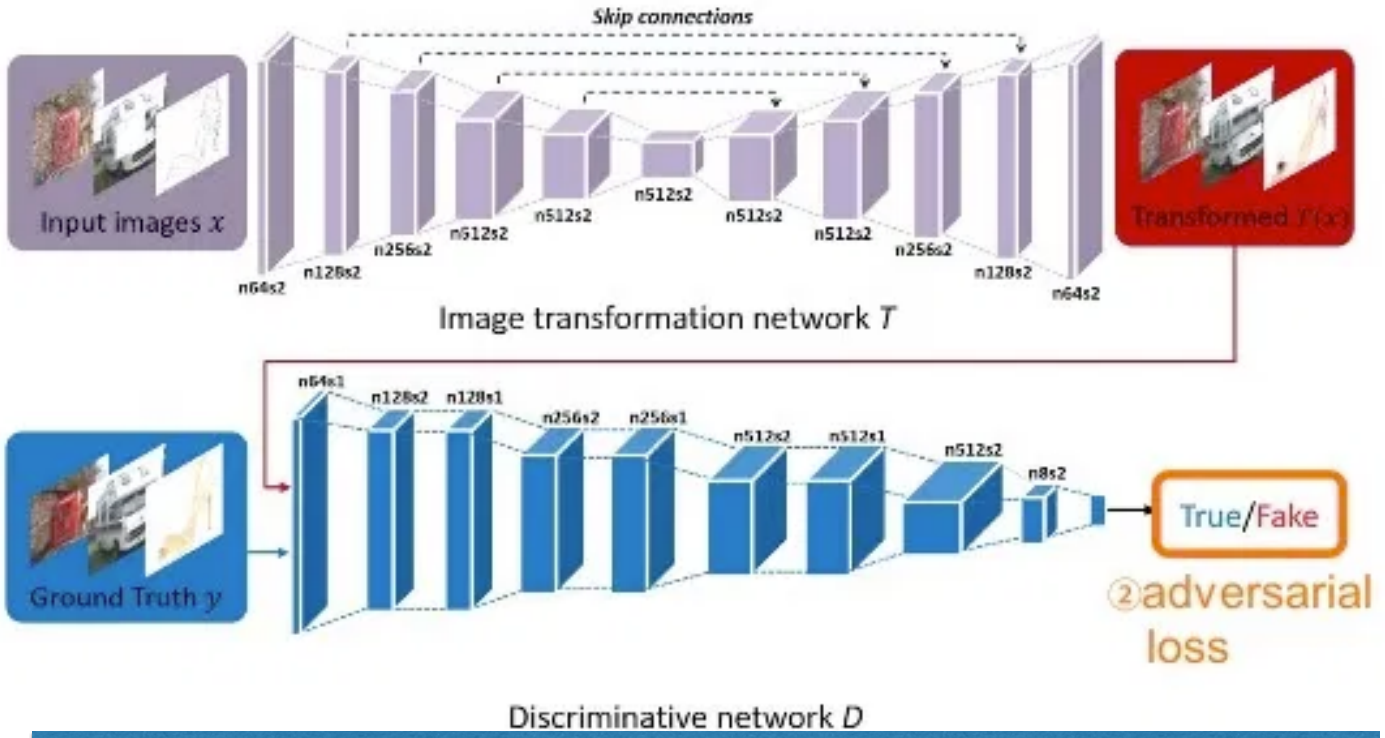
GAN Variations

- Deep Convolutional GAN (DCGAN)
  - Convolutional layers to generate image
  - Can be used to generate photorealistic images



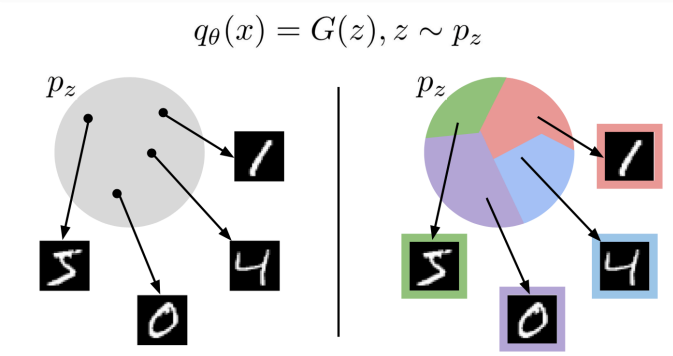
GAN Variations

- Conditional GAN (cGAN)
  - Can be used to translate from one domain (source) into another (target)
  - In addition to the random noise, sample from source domain is fed into generator



GAN Variations

- Wasserstein GAN (WGAN)
  - Different loss function than minimax (the one I showed so far)
  - Help address the issue of mode collapse issue during GAN training
    - Generator may collapse to a setting where it always produces same outputs (gets stuck in a small space with low variety), this is a common failure case for GANs
    - fails to learn to represent the complex real-world data distribution



Today's Topics

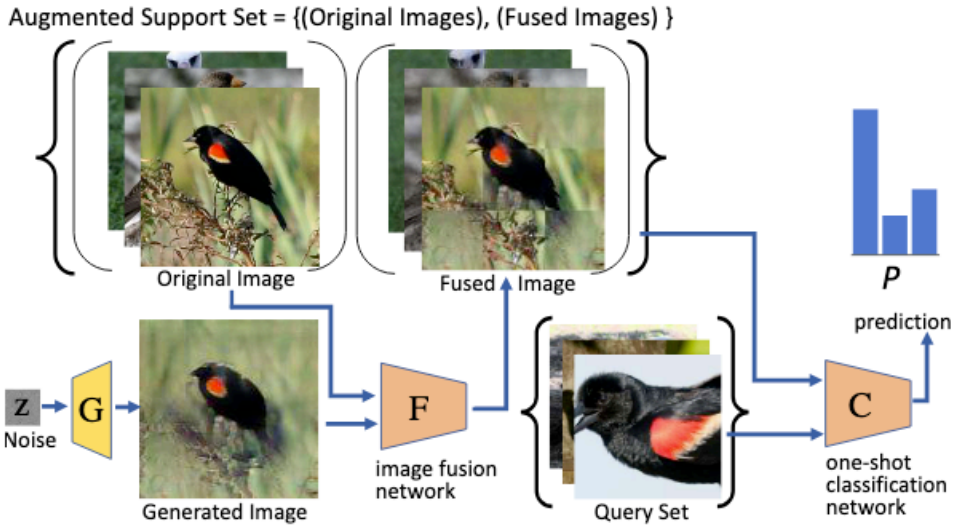
- **Generative Modeling**
- Introduction to generative adversarial network (GAN)
- GAN variations
- **Computer vision applications with GAN**
- GAN model in my research

Why GAN is useful?

- Image generation eg, StyleGAN, change the content/style of the image



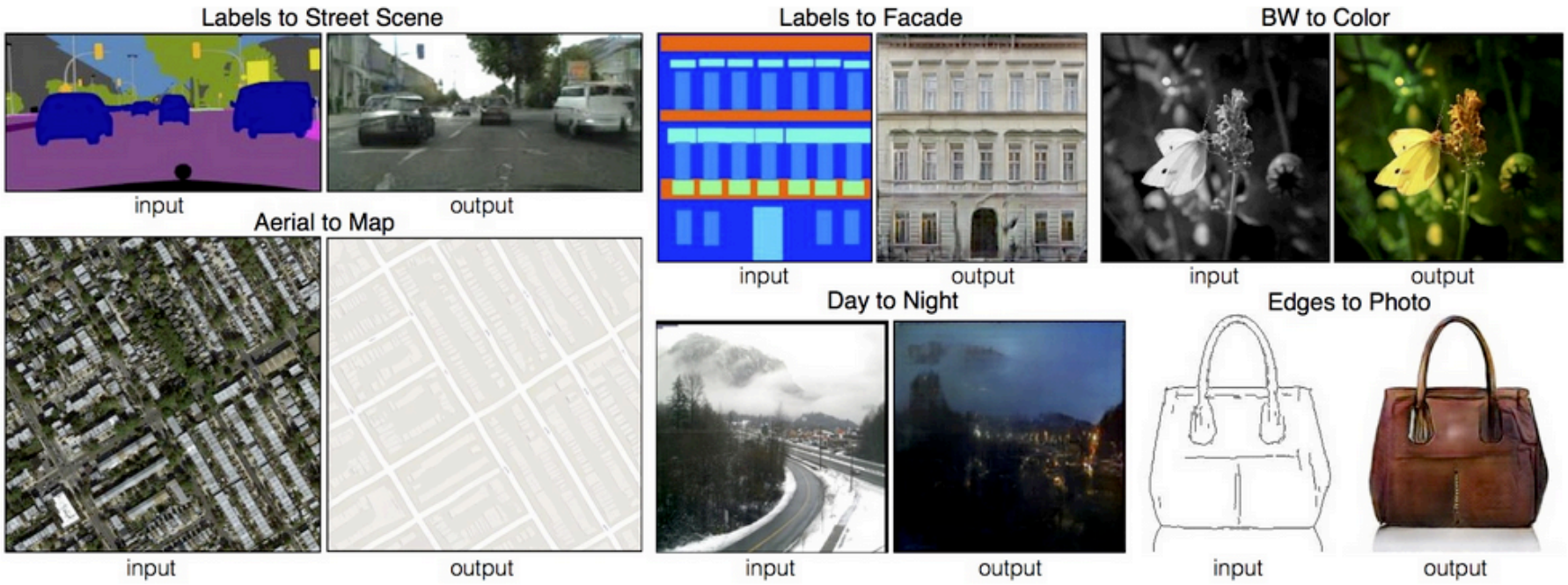
- Data augmentation
  - Generate synthetic data
  - improve your downstream models with GAN-generated data



Computer Vision Applications

- Paired Image to image translation (pix2pix)
  - Add extra loss function in addition to minimax
  - U-Net generator
  - PatchGAN discriminator

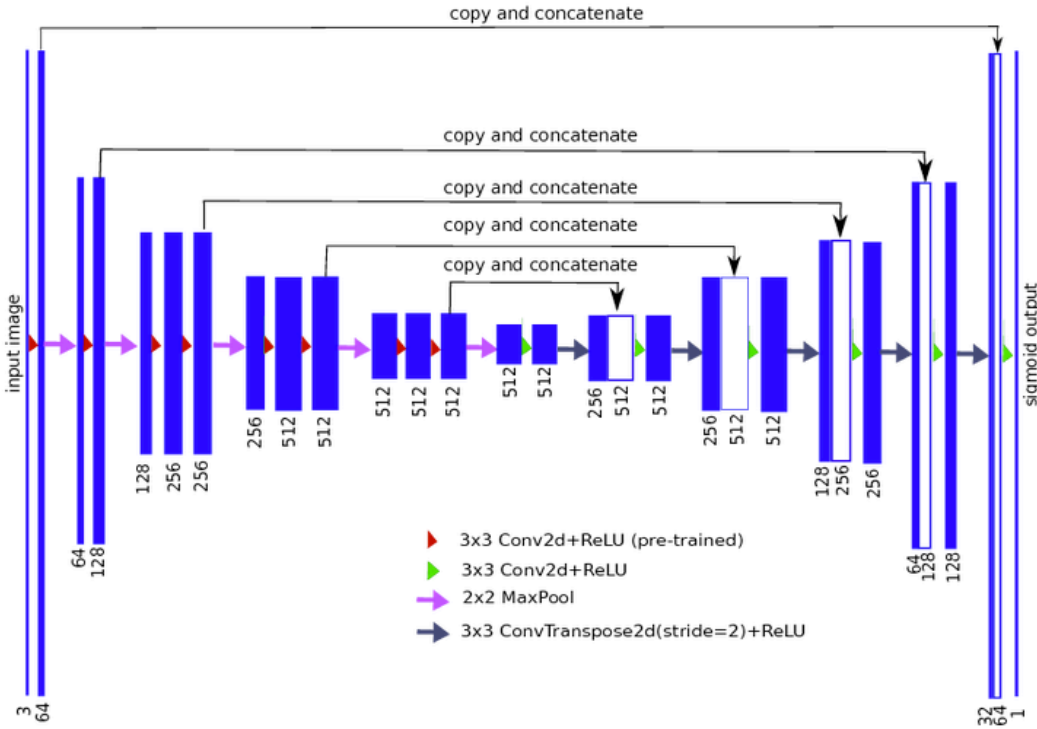
$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G).$$



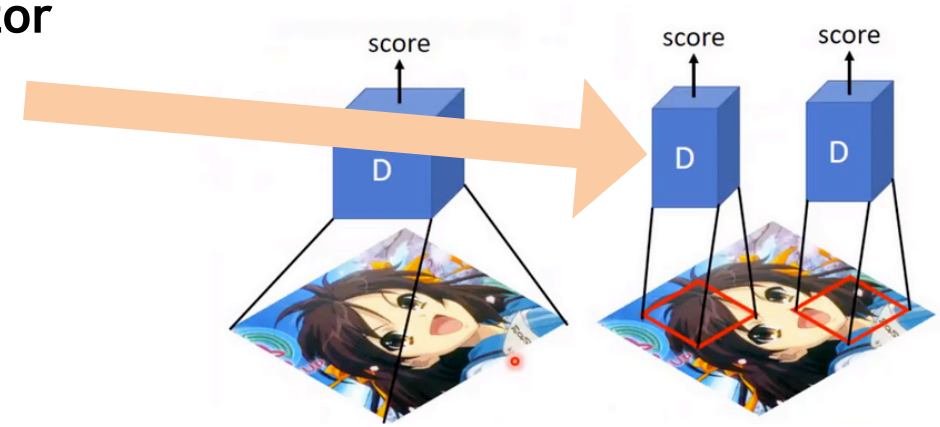
Example results on several image-to-image translation problems. In each case we use the same architecture and objective, simply training on different data.

Computer Vision Applications

- Pix2pix
  - U-Net **generator**

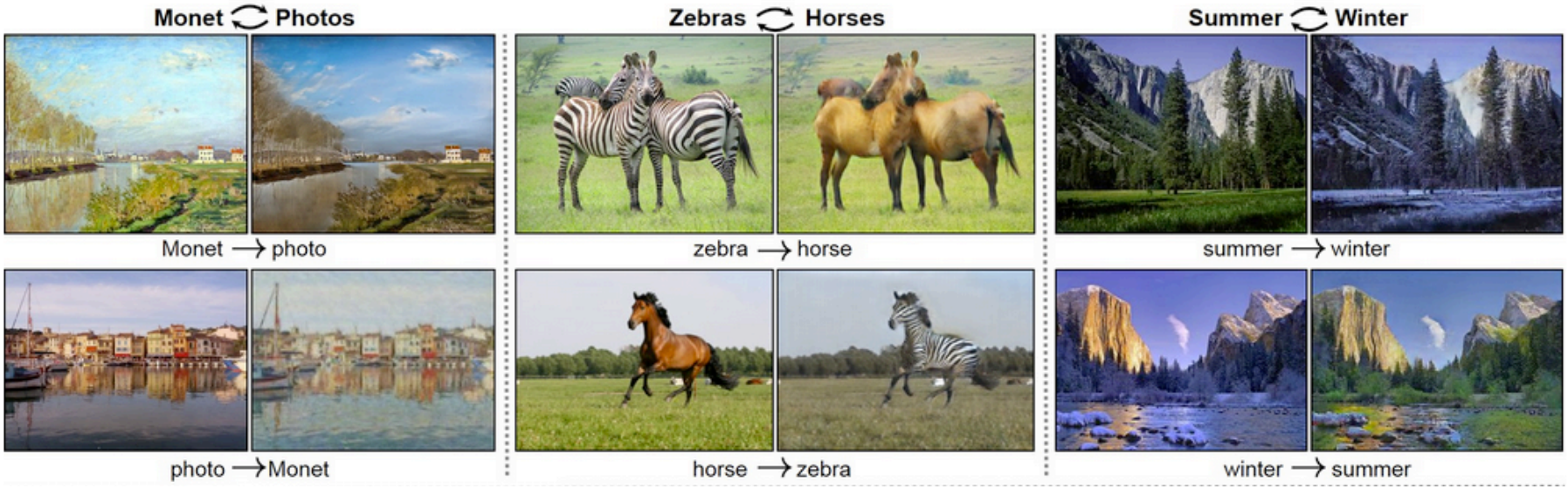


  - PatchGAN **discriminator**



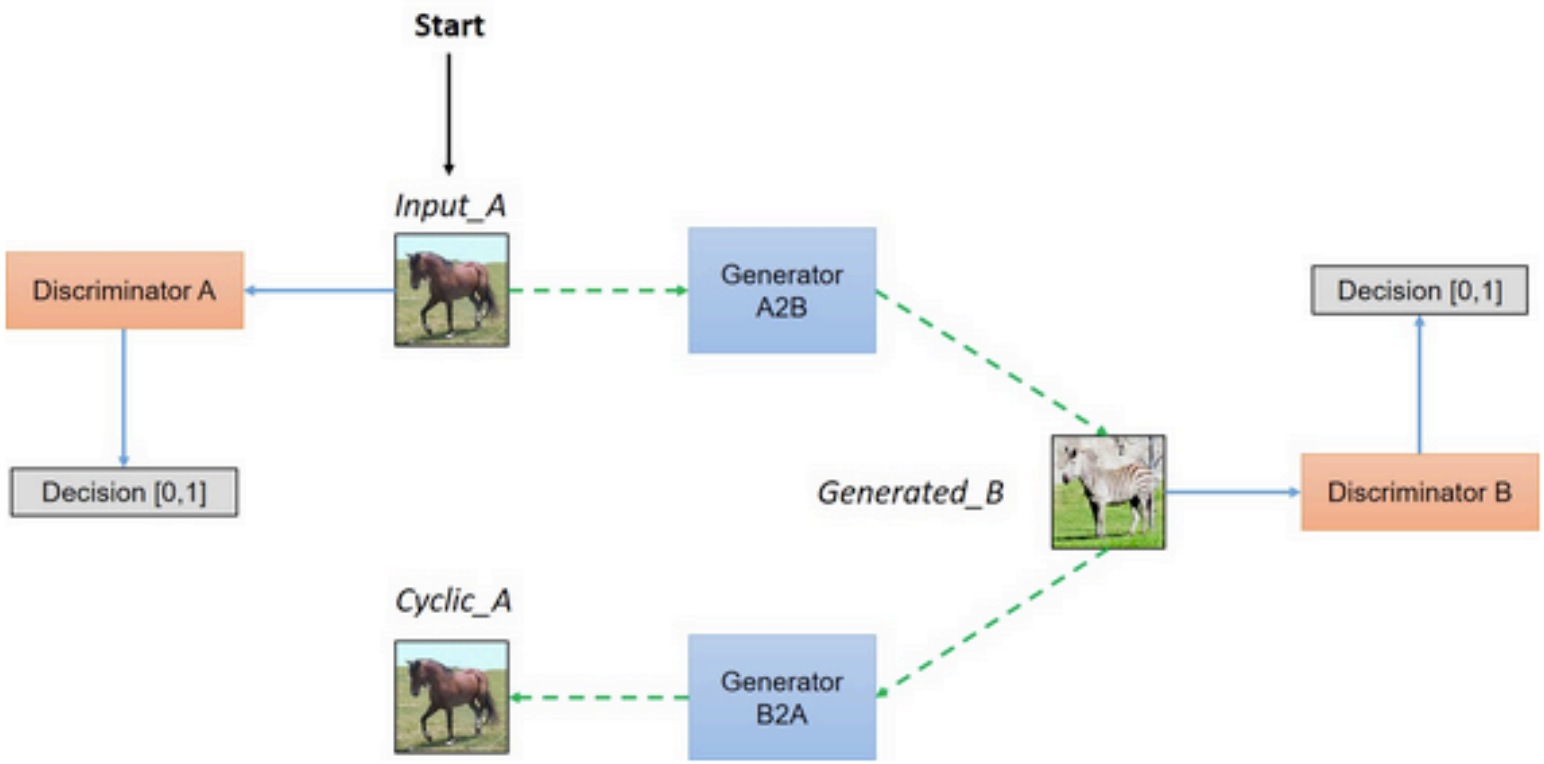
Computer Vision Applications

- Unpaired image-to-image translation (CycleGAN)
  - an unpaired image-to-image translation model, to adapt horses to zebras (and vice versa) with two GANs in one



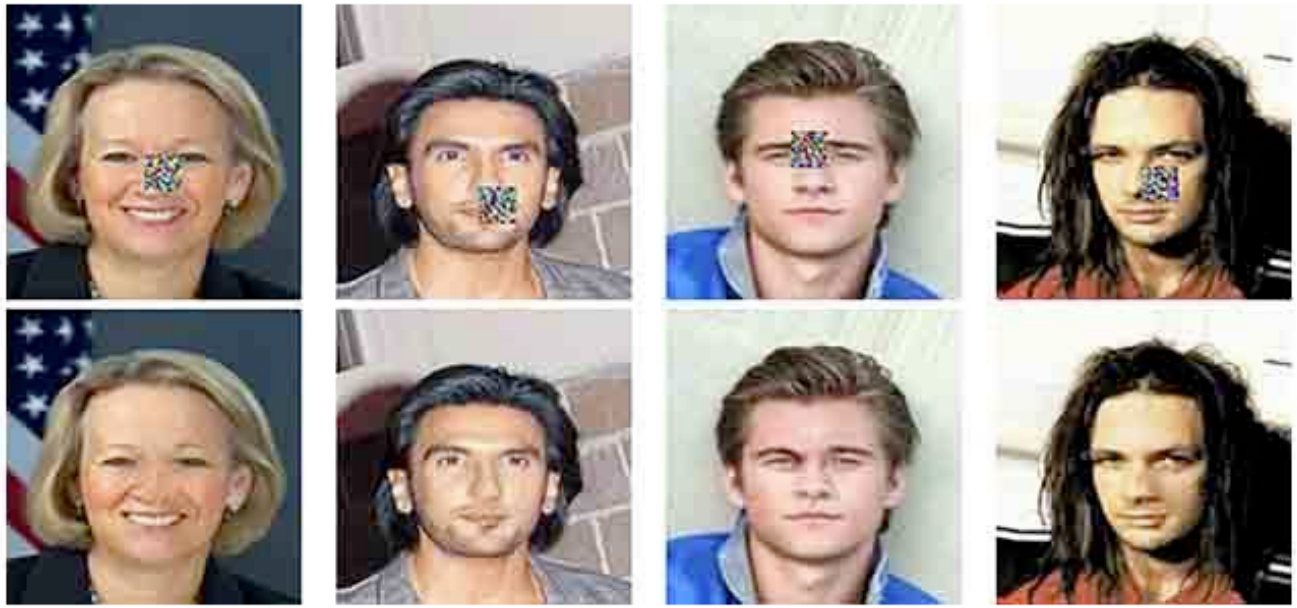
Computer Vision Applications

- Unpaired image-to-image translation (CycleGAN)
  - an unpaired image-to-image translation model, to adapt horses to zebras (and vice versa) with two GANs (A2B and B2A)



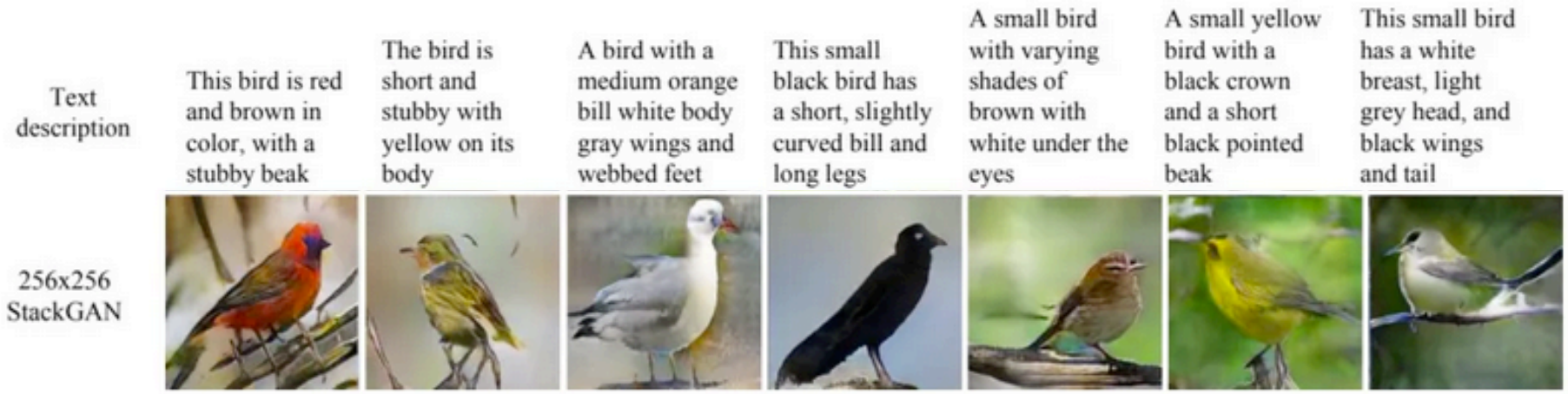
Computer Vision Applications

- Face Inpainting
  - Filling the masked region of a real image



Computer Vision Applications

- Text to Image (StackedGAN)
  - Generating an image from a textual description



Today's Topics

- **Generative Modeling**
- Introduction to generative adversarial network (GAN)
- GAN variations
- Computer vision applications with GAN
- **GAN model in my research**

Depth Estimation from an Image

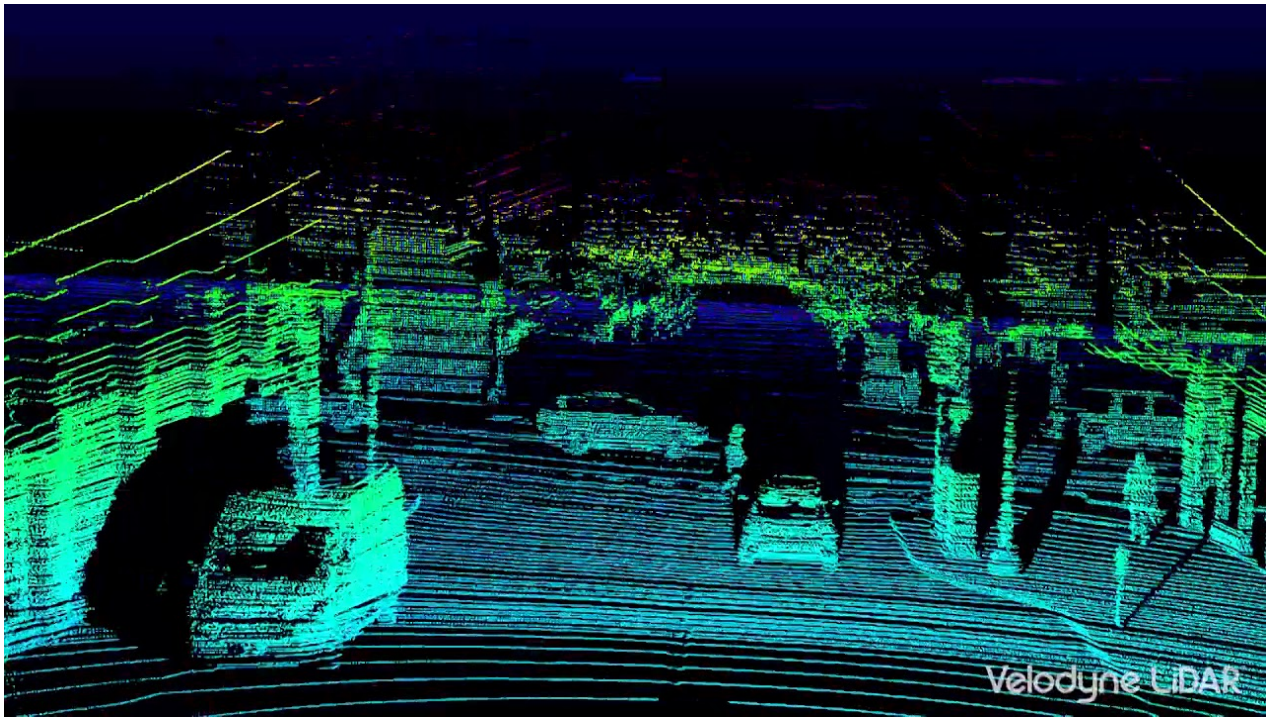
- **Motivation**
  - Outdoor image may contain object at a far-away distance
  - Useful for an autonomous robot to find depth of a distant object
  - object detection in military mission
  - surveillance
  - autonomous driving etc



Figure: Long-range surveillance

Depth Estimation from an Image

- **Difficulties with existing approaches**
  - Most of monocular depth estimation algorithms provide depth estimates less than 100 meters
  - Depth sensors (eg, Laser, LiDAR) have limited range:



Depth Estimation from an Image

- **My solution:** GAN based algorithms that can predict the depth from an image

Figure: RGB and generated Depth images in the city of Houston, Texas





- Developed an RGB and Depth image pair generation algorithm that goes up to 1000 meters
  - utilizing 3D reconstruction of the scene from multi-view images

Depth Estimation from an Image

- **Conditional Generative Adversarial Network (cGAN*):**
  - Depth prediction as an image-to-image translation problem using cGAN, where the goal is to translate an RGB image (source domain) to a depth image (target domain)

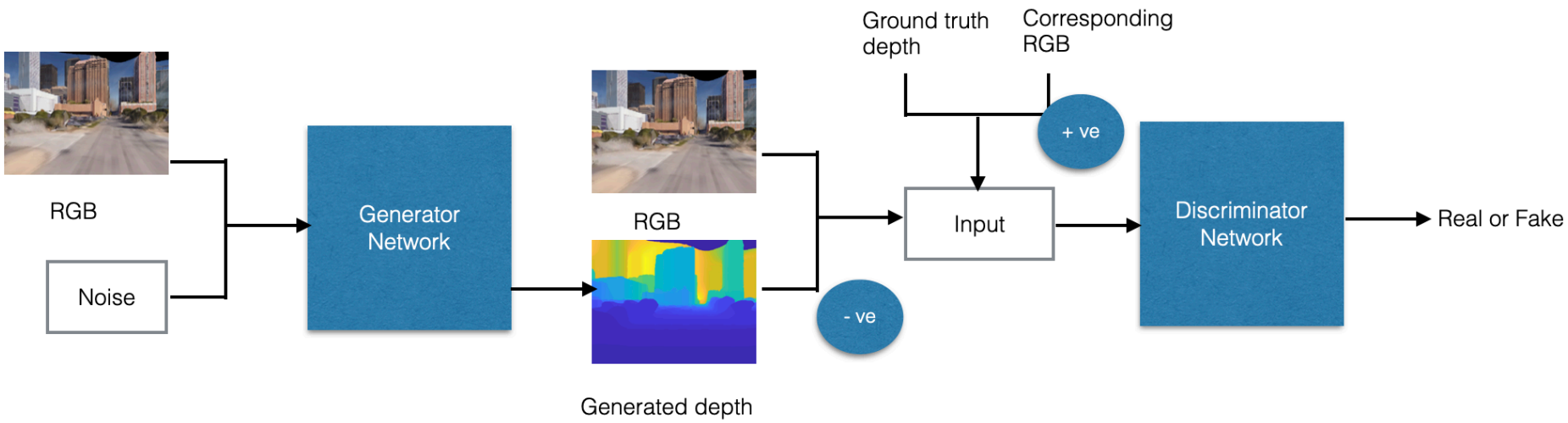


Figure: cGAN architecture for depth prediction

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

Synthetic Data Generation

- Collect urban images of a geolocation (eg, New York City, Chicago, Houston) using Google Earth software
- Reconstruct the 3D model from the images
- Annotate key points for the trajectory along the model
- Render the RGB+ Depth image pairs along the trajectory

Image Collection for 3D Reconstruction

- Extract images (street view) of a city eg, NYC from multiple viewpoints
- Google Earth allows to record a video-clip of a city as user hover around from different viewpoints
- Extracted images are sampled for 3D reconstruction using *Adobe Remake*

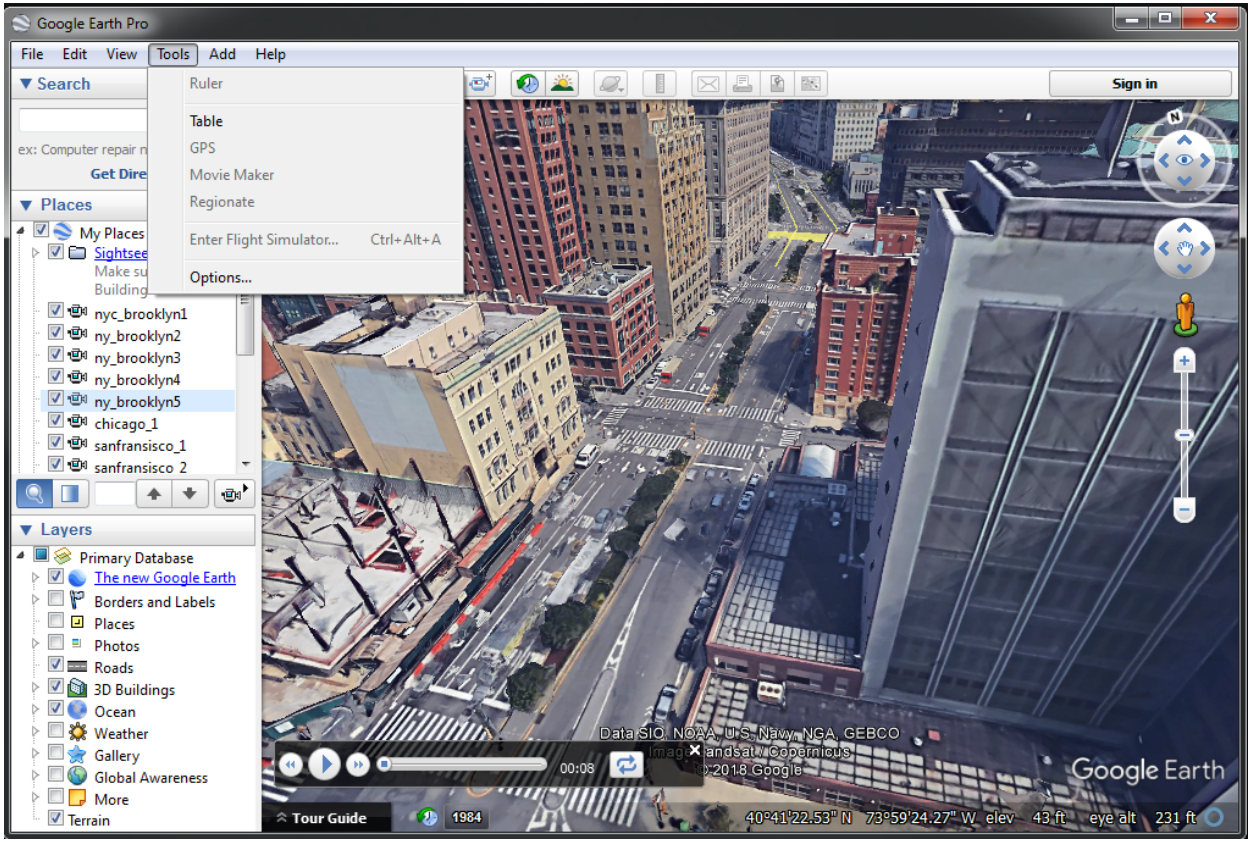


Figure: Collecting a video clip from Google Earth Pro in New York City. The *Movie Maker* feature enables this option. Images are exported from the recorded video clip at 30 fps.

3D Reconstruction using Adobe Remake

- Model of the city is reconstructed from a subset of manually selected images (120~250)

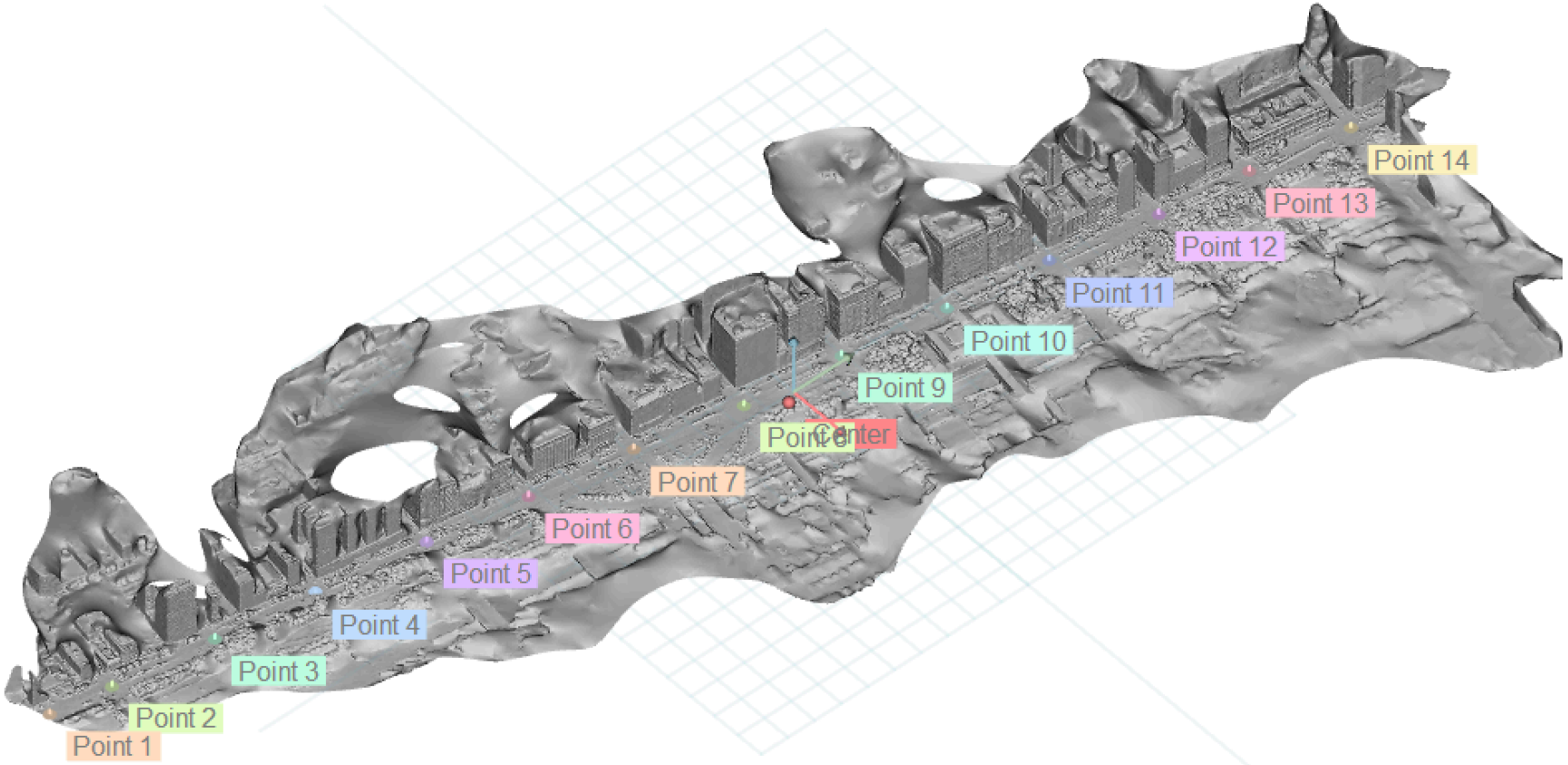


Figure: An example 3D reconstructed model from our dataset. The keypoints (*Point 1*, *Point 2*, *Point 3*, etc.) are manually selected in the 3D model.

Render an RGB and Depth pairs from 3D Model

- Generate a trajectory of a moving virtual camera between two keypoints *A* and *B* in the 3D model
- Render the scene from the viewpoint of the virtual camera using OpenGL from each point along the trajectory

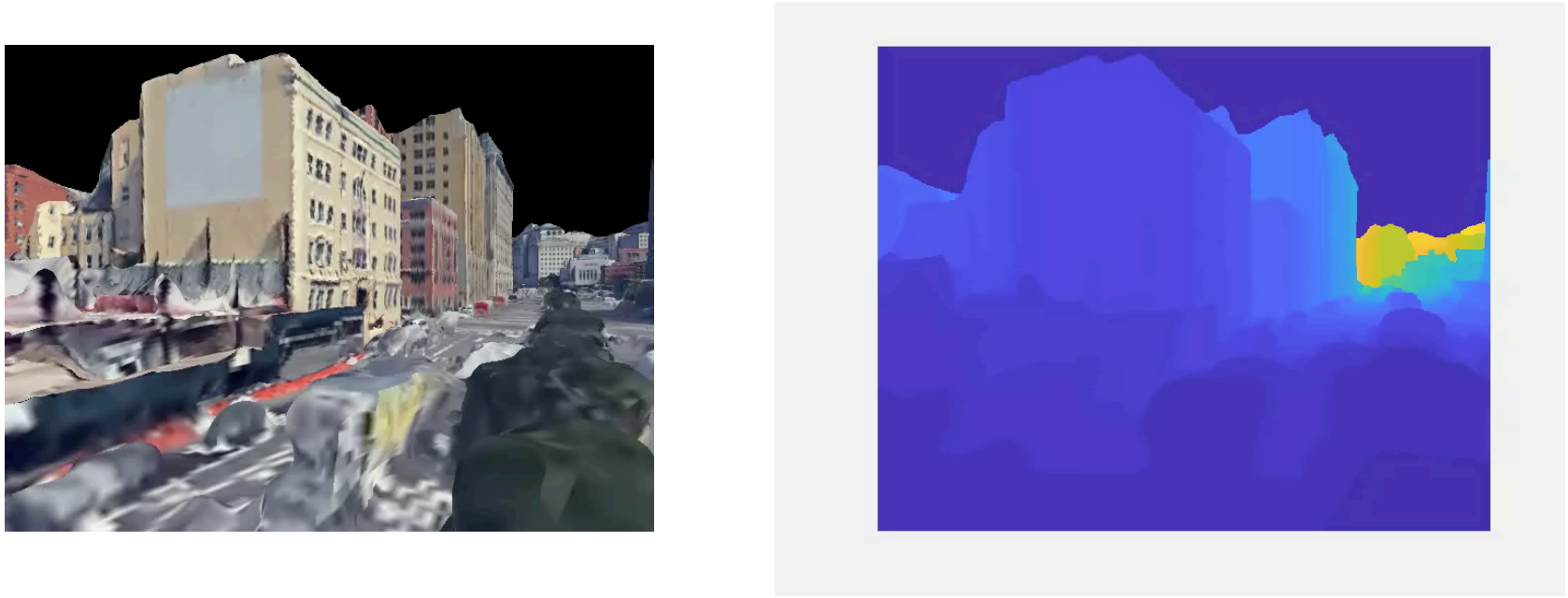
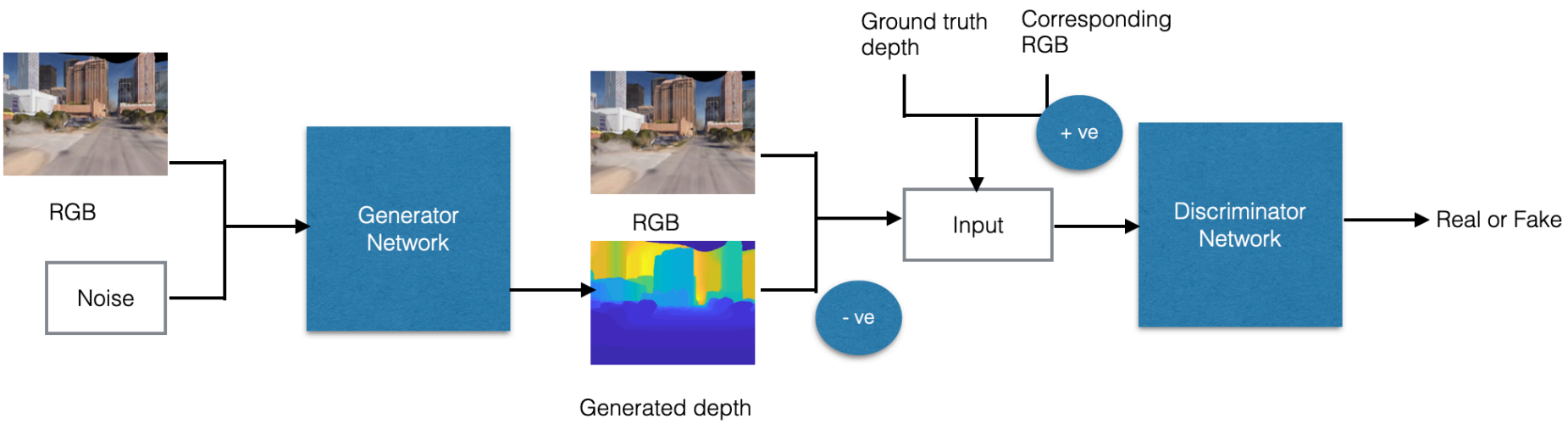


Figure: Rendered RGB and Depth images from different virtual camera trajectory locations inside the *New York City* reconstructed 3D model.

cGAN trained using rendered depth images

- Trained cGAN using the quantized depth values as ground truths



- Evaluated depth prediction on 3 metrics:
  - Absolute Relative Error
  - Linear RMSE
  - Scale Invariant RMSE

Scene	# Frames
New York City	300
Miami	250
Houston	250
Salt Lake City	100
Total	900

Visualization of the Depth Prediction

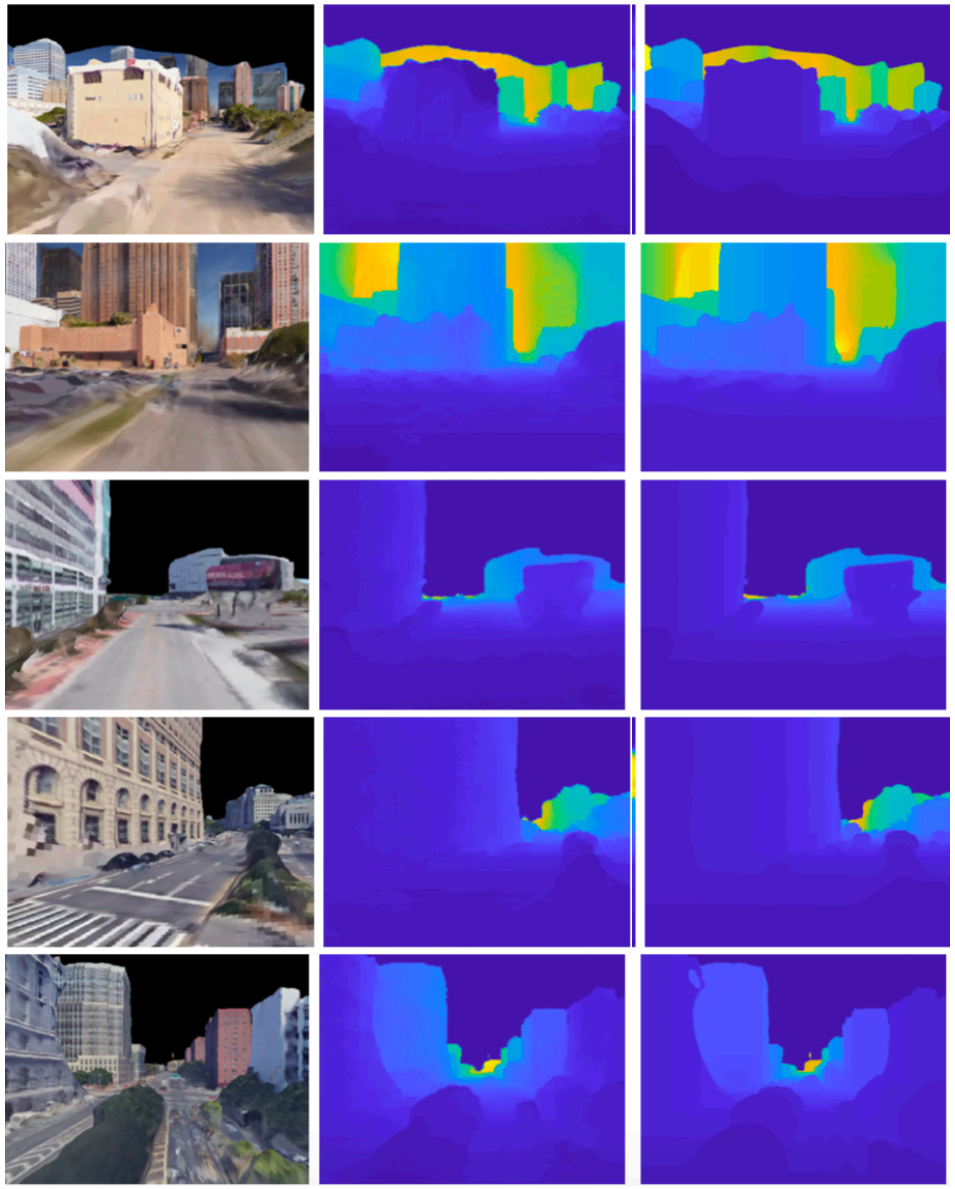


Figure: (From left to right) the input RGB image, the prediction from GAN model, and the ground truth depth.

Next Lecture
Generative Model: VAE (likelihood-Based Model)

- Generative modeling is based on representation of probability distribution

- Likelihood-based models:
  - directly learns the distribution's probability density function (PDF) via maximum likelihood estimation method

  - Variational Autoencoder (VAE) is an example of likelihood-based generative model

  - Limitations
    - rely on surrogate objectives to approximate maximum likelihood training
    - require strong restriction on the model architecture for tractable normalization