Deep Learning (1470)

Randall Balestriero

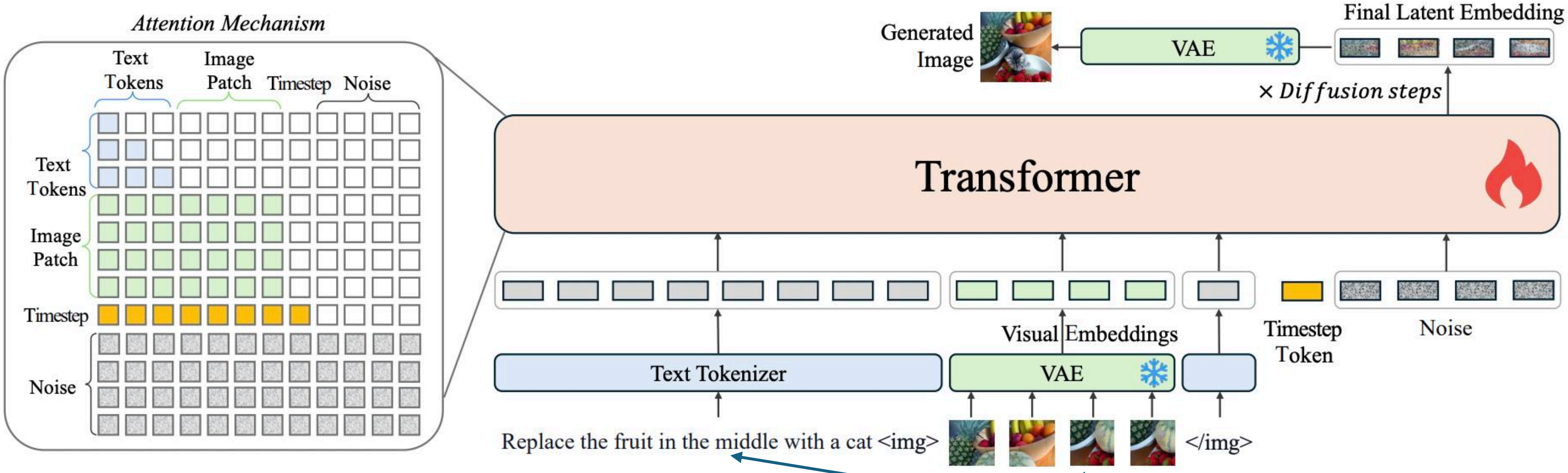
Class 16: GANs

Recap!

What *might* this look like?

Output image

Output embedding vector



Take in multiple modes of input (text, images, etc.)
(Multi-modal learning)

Source: OmniGen: Unified Image Generation: https://arxiv.org/pdf/2409.11340

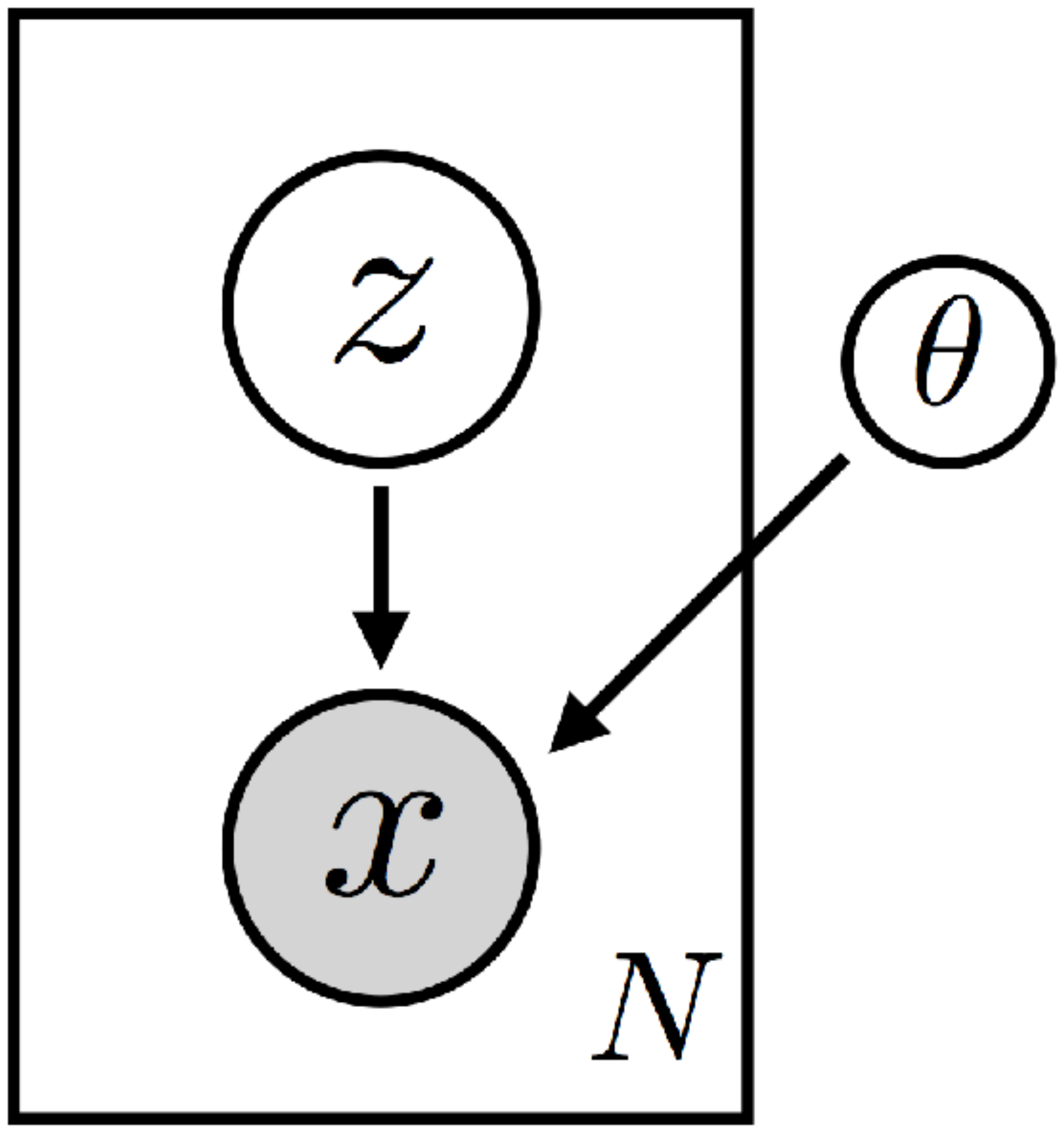
GANs

GANs

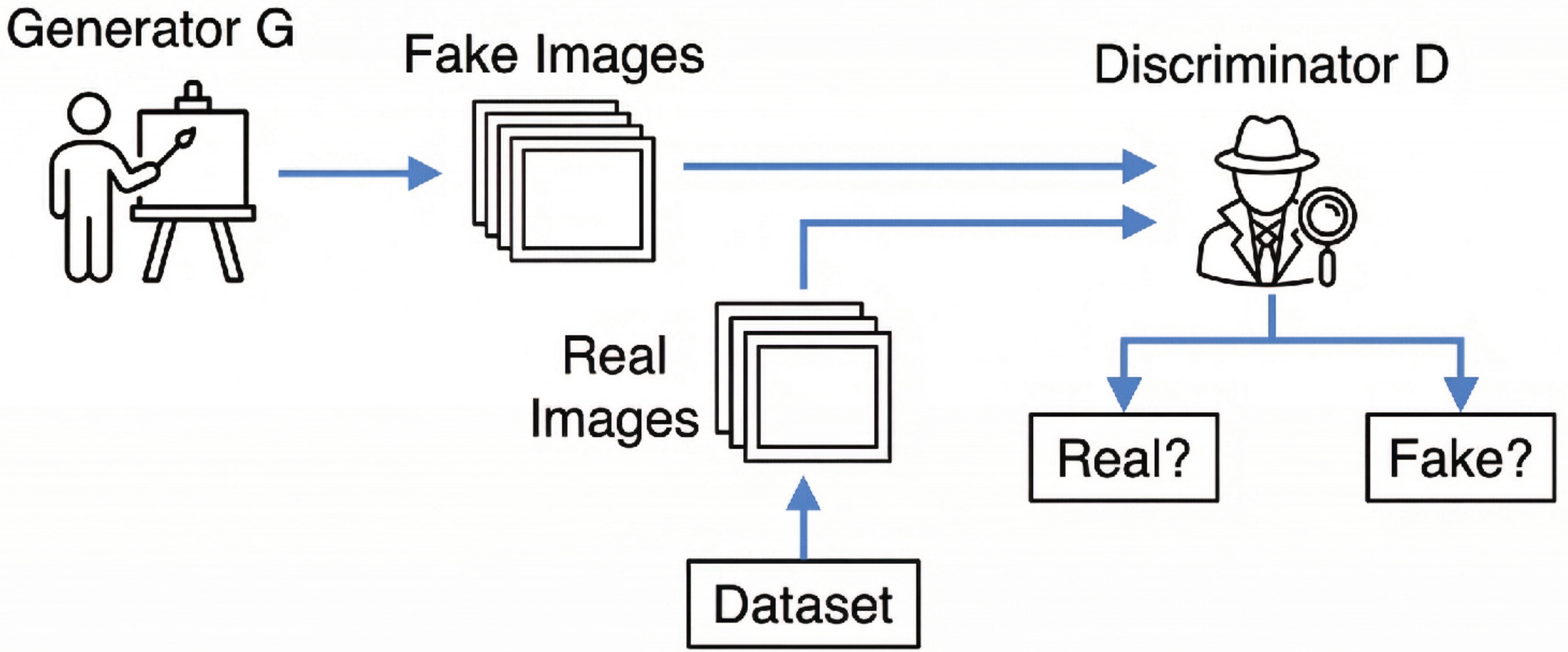
Generative Adversarial Networks

GANs

- Given “nothing”
- Sample a random gaussian vector z
- Generate an image via decoder(z) that looks like a real image

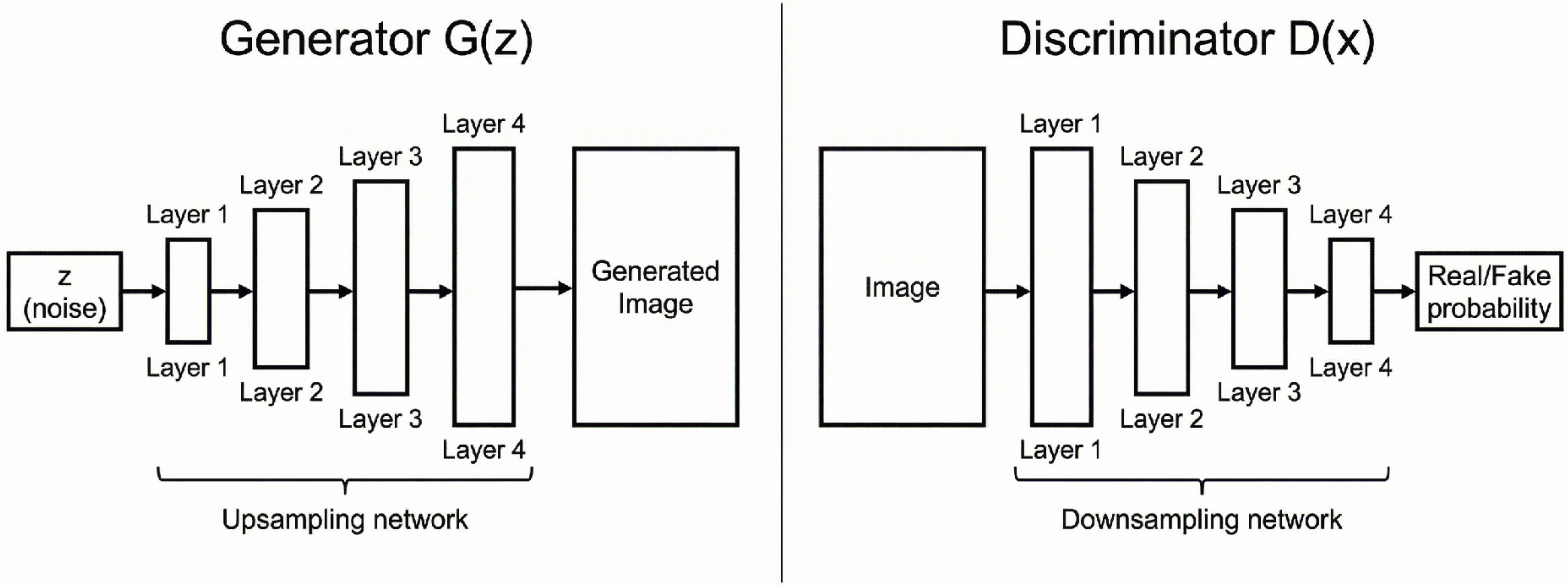


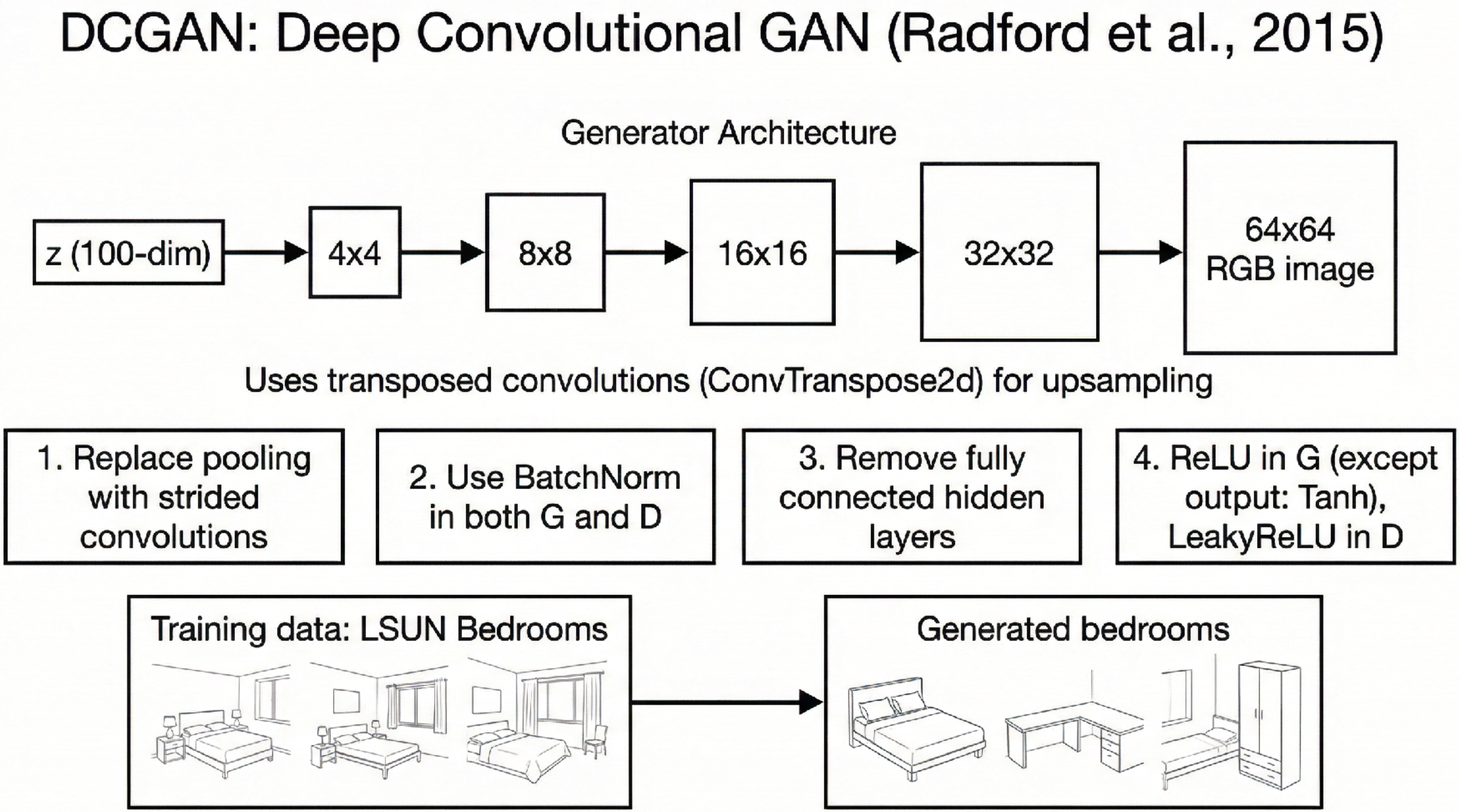
Generative Adversarial Networks: The Core Intuition



Two networks compete: Generator creates, Discriminator judges.
Both improve through competition.

GAN Architecture: Generator & Discriminator




DCGAN: Deep Convolutional GAN (Radford et al., 2015)
Generator Architecture
z (100-dim)
4x4
8x8
16x16
32x32
64x64
RGB image
Uses transposed convolutions (ConvTranspose2d) for upsampling
1. Replace pooling with strided convolutions
2. Use BatchNorm in both G and D
3. Remove fully connected hidden layers
4. ReLU in G (except output: Tanh), LeakyReLU in D
Training data: LSUN Bedrooms
Generated bedrooms

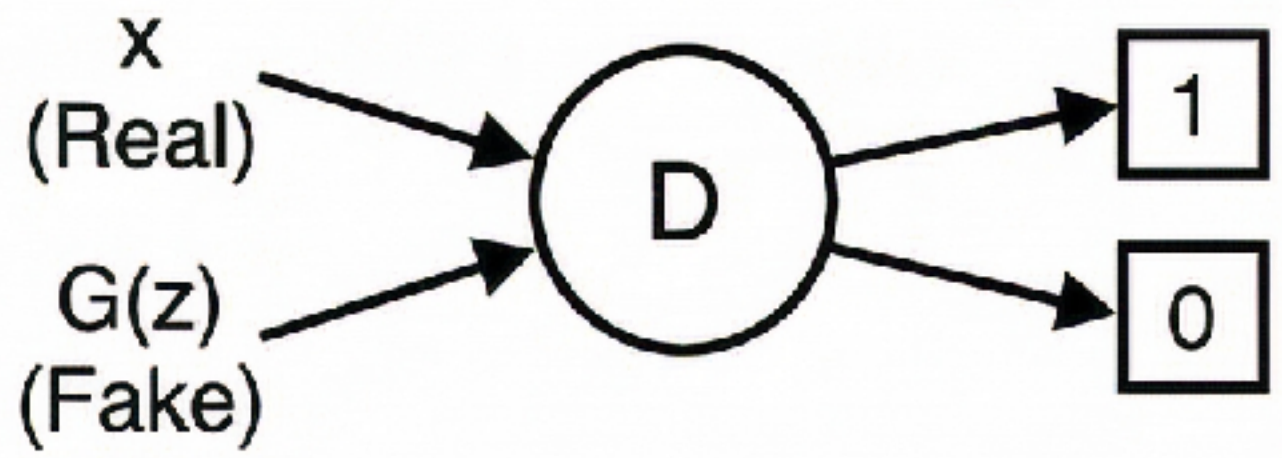
GAN Loss Functions

Discriminator Loss

$$\boxed{\mathcal{L}_D = -\mathbb{E}[\log D(x)] - \mathbb{E}[\log(1 - D(G(z)))]}$$

Goal: Maximize probability of correct classification

D wants to output 1 for real, 0 for fake

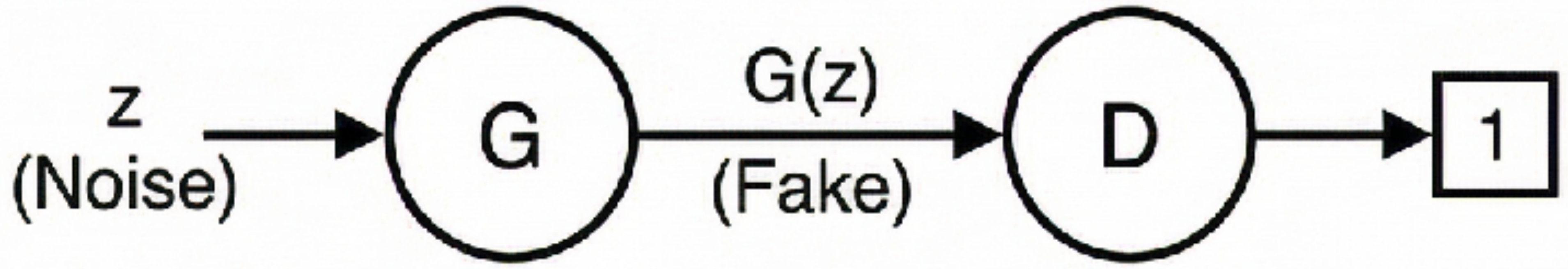


Generator Loss

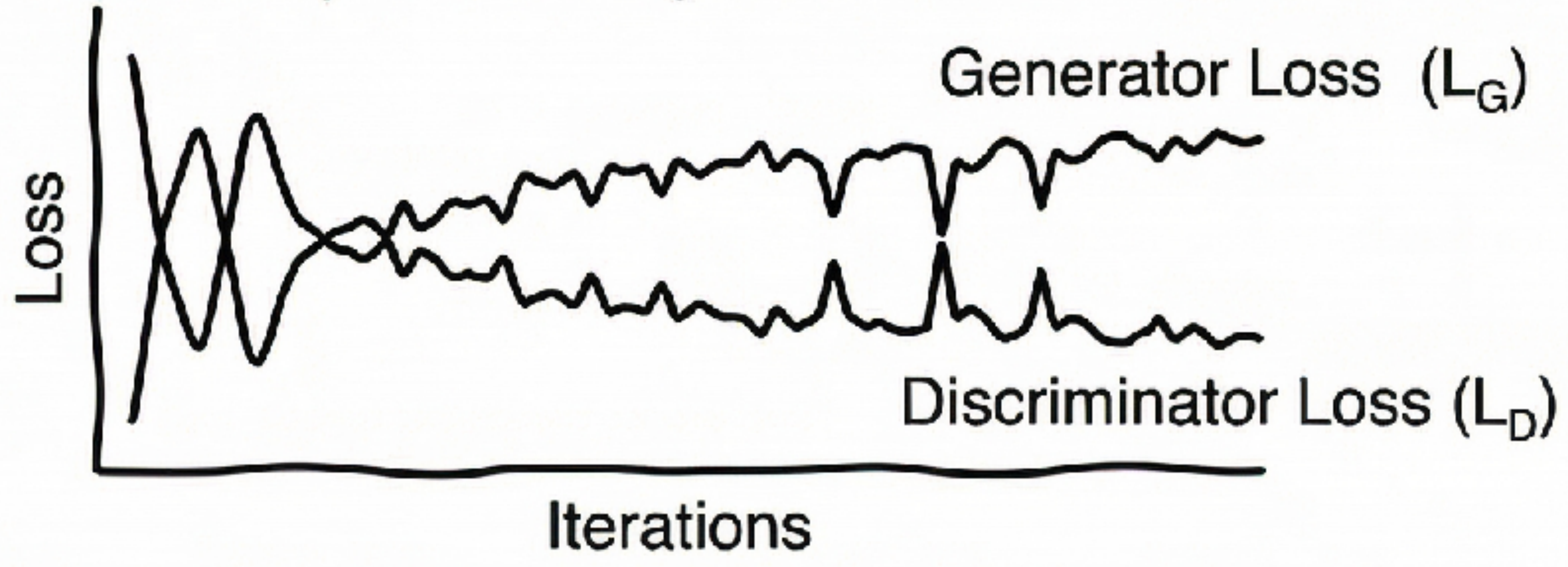
$$\boxed{\mathcal{L}_G = -\mathbb{E}[\log D(G(z))]}$$

Goal: Fool discriminator into outputting 1

G wants D(G(z)) to be close to 1



Simple Training Curve Sketch

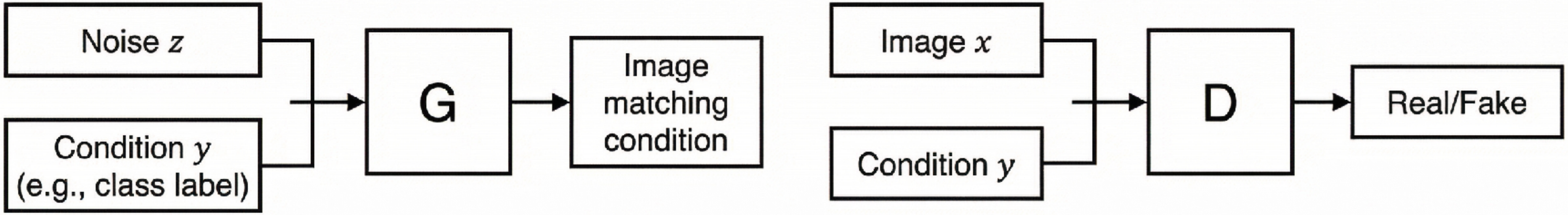


At **equilibrium:** D outputs 0.5 for all inputs

How to “control” the generation?

Conditional GAN (cGAN): Controlled Generation

TOP: Architecture diagram:

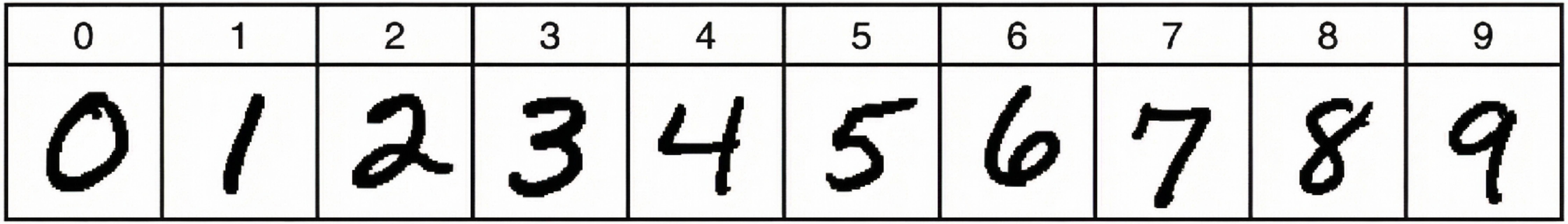


Key equations:

$G(z, y) \rightarrow$ image conditioned on y

$D(x, y) \rightarrow$ probability that x is real AND matches y

Visual example (MNIST style):

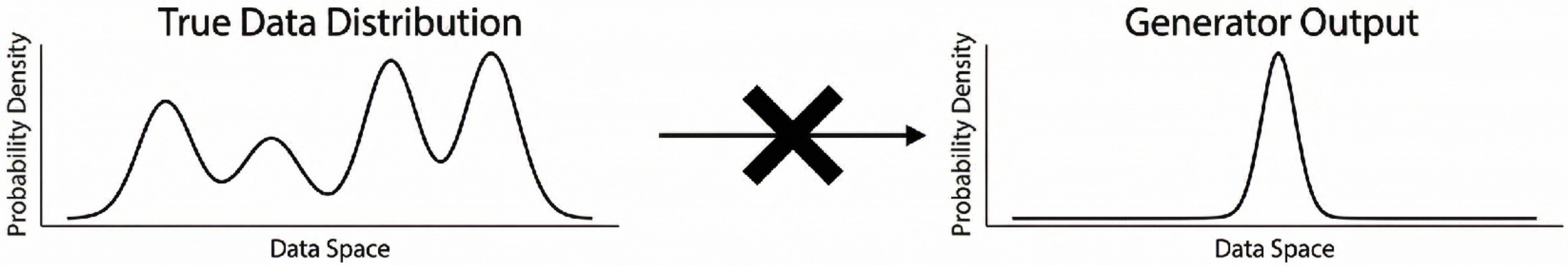


Same architecture, different condition → different output

Can you think of a problem?

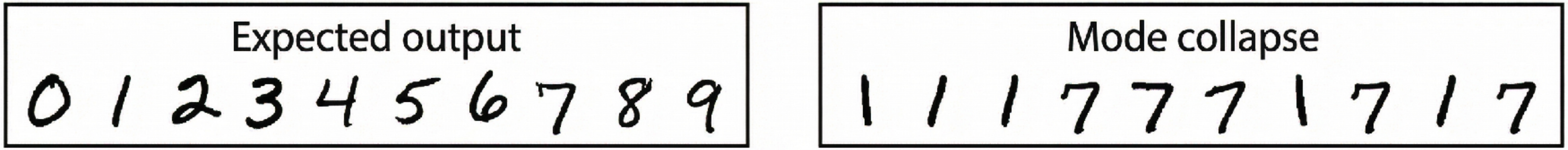
Problem 1: Mode Collapse

Visual Explanation with Distributions



Generator ignores most of the data distribution

Concrete MNIST Example



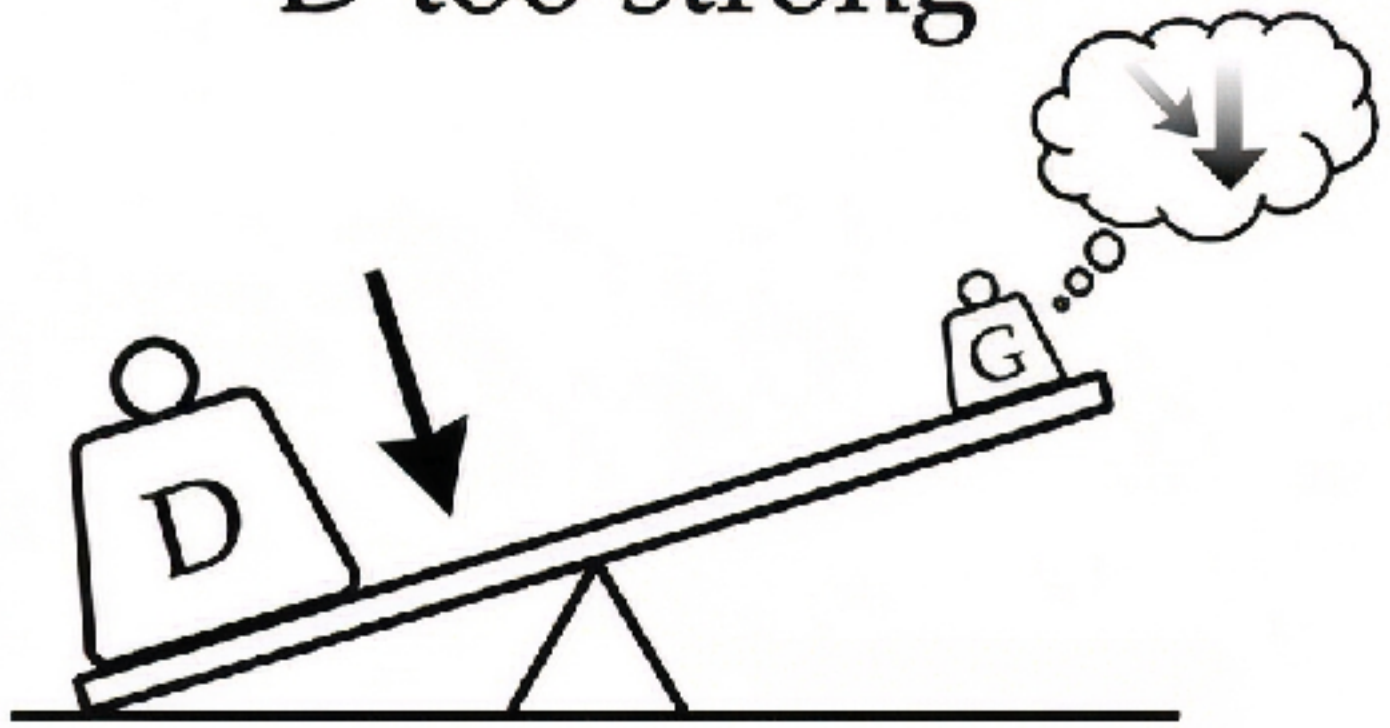
Generator only produces a few digit types

Why it happens

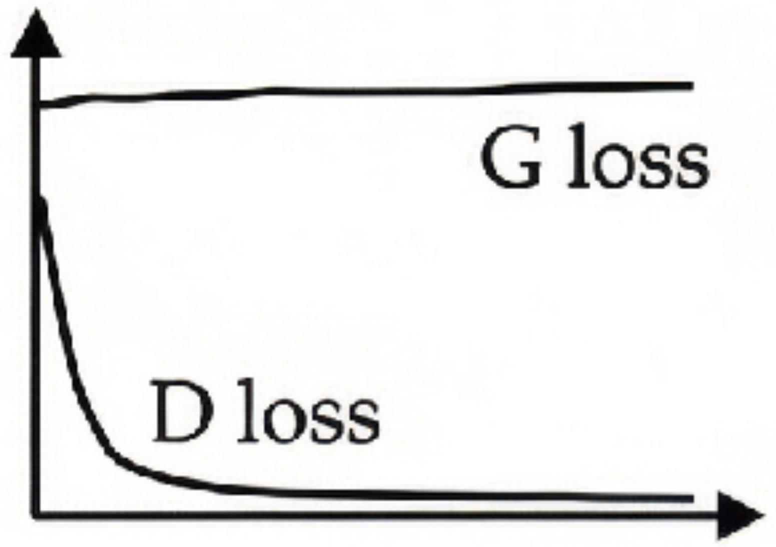
1. G finds one output that fools D → 2. G exploits this rather than diversifying → 3. No explicit diversity term in loss

Problem 2: Training Instability

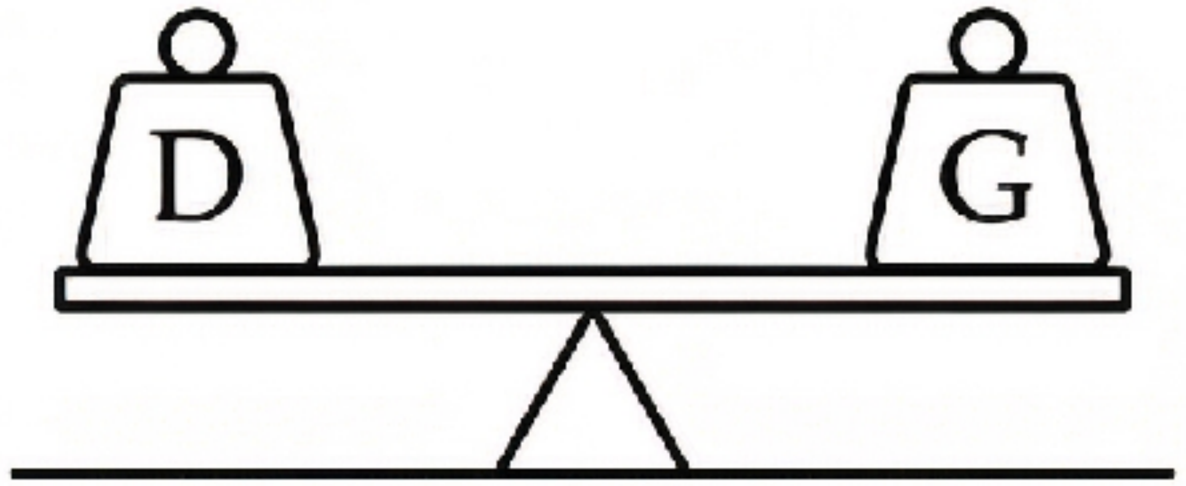
D too strong



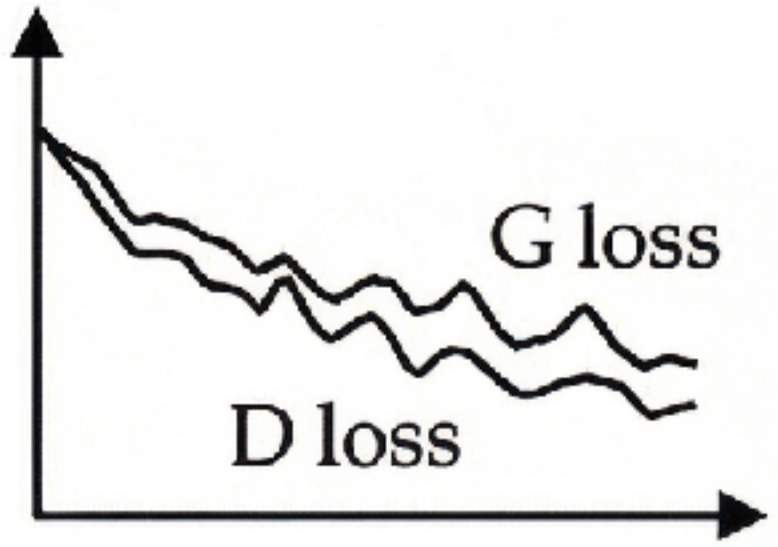
D wins easily, G gets vanishing gradients



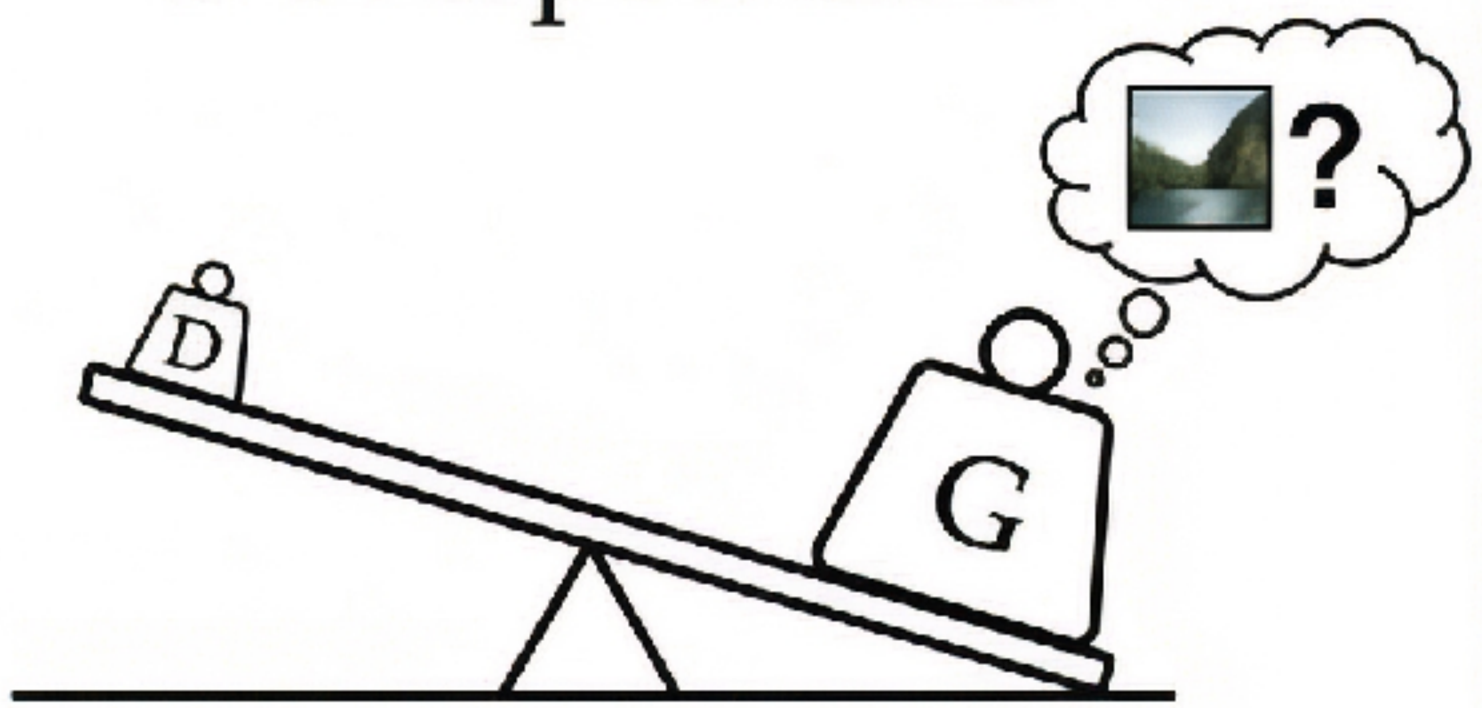
Balanced (ideal)



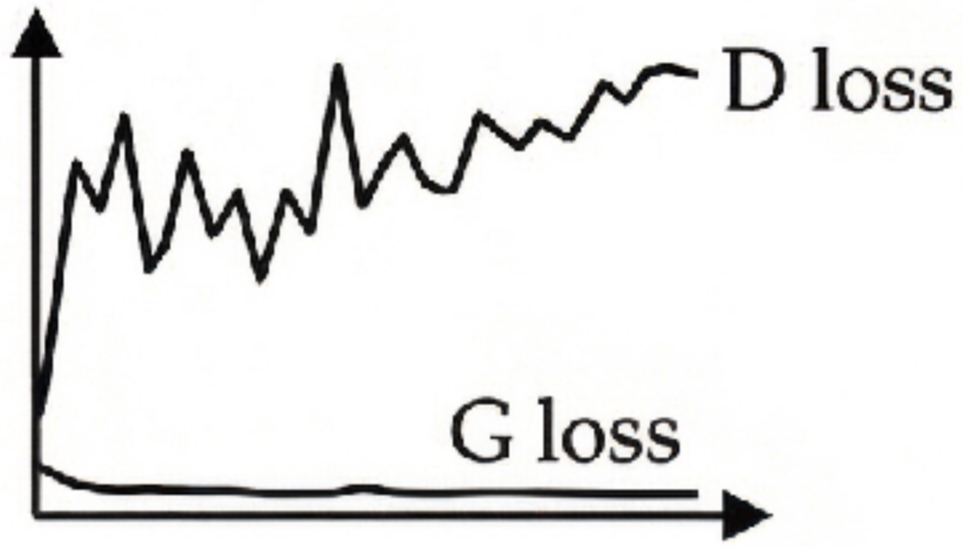
Healthy competition



G overpowers D



G fools D completely, no learning signal

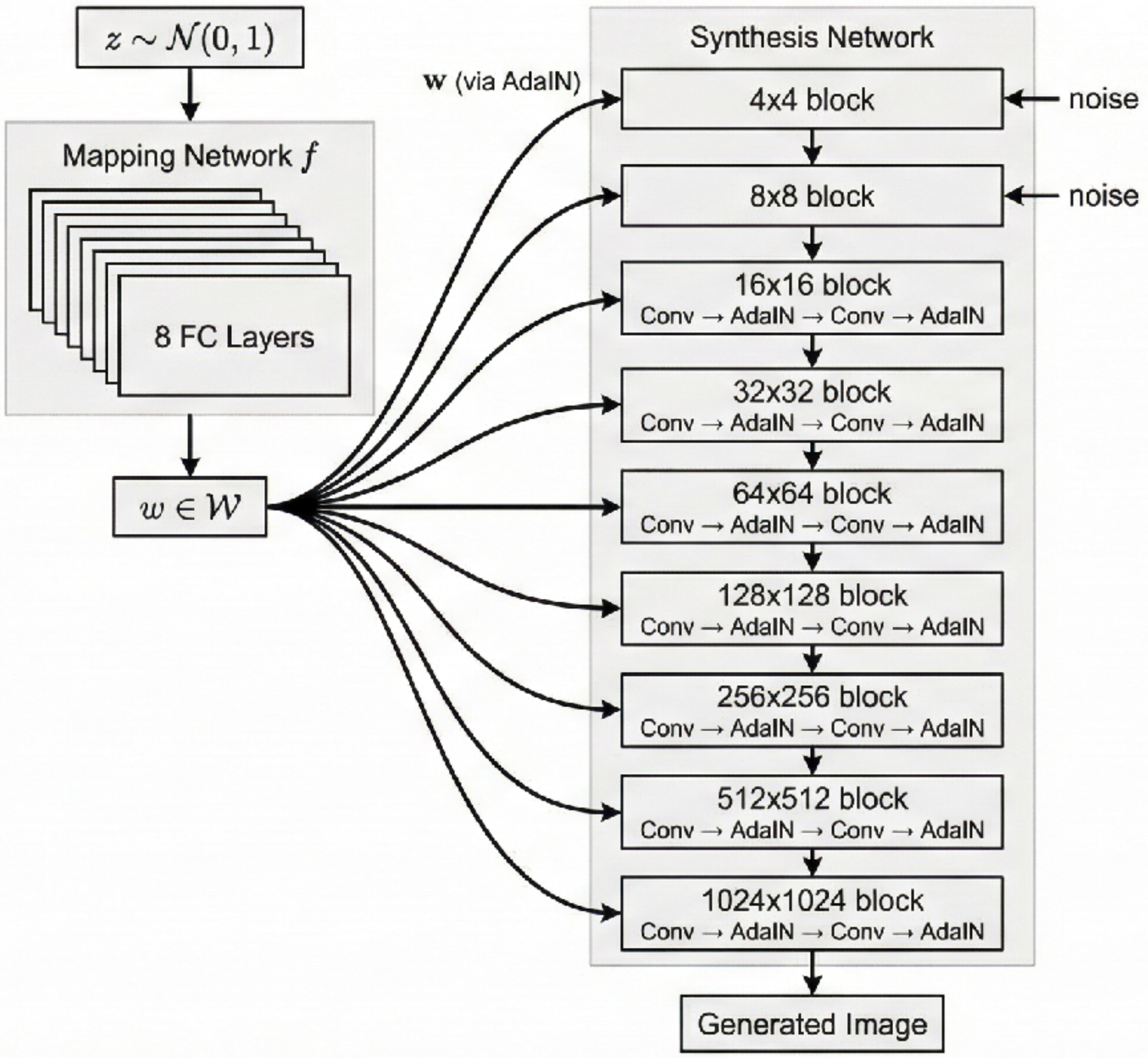


Common symptoms list:

- 1. Losses oscillate wildly, never converge
  2. Generator produces noise or artifacts
- 3. Discriminator outputs 0 or 1 for everything
- 4. Training suddenly diverges

StyleGAN: Style-Based Generator (Karras et al., 2018)

Detailed Architecture



Key Components

AdaIN (Adaptive Instance Norm)

$$\mathrm{AdaIN}(x, y) = y_s * \frac{x - \mathrm{mean}(x)}{\mathrm{std}(x)} + y_b$$

Style y controls mean and variance of features

Losses Used

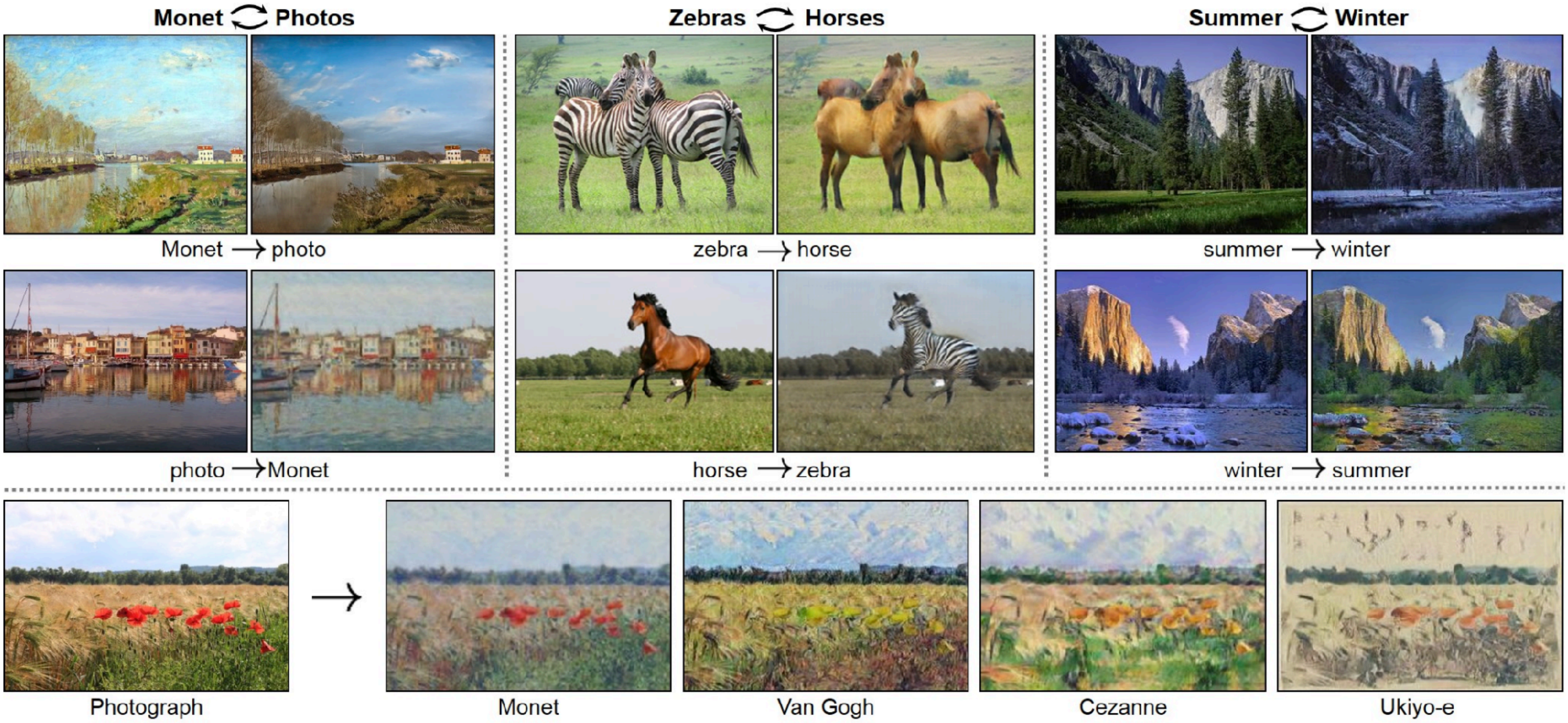
- Adversarial loss (non-saturating)
- R1 regularization on D
- Path length regularization
- Style mixing regularization

Progressive Training

Start at 4x4, gradually add higher resolution layers

Is it Real?

CycleGAN: Unpaired Image-to-Image Translation
discriminates real vs fake horses
D_X
Domain X
Horses (x)
$G: X \to Y$
transforms horse to zebra
$F: Y \to X$
transforms zebra to horse
Domain Y
Zebras (y)
D_Y
discriminates real vs fake zebras
Adversarial Loss
$\mathcal{L}_{\text{GAN}}(G, D_Y) = \mathbb{E}[\log D_Y(y)] + \mathbb{E}[\log(1 - D_Y(G(x)))]$
$\mathcal{L}_{\text{GAN}}(F, D_X) = \mathbb{E}[\log D_X(x)] + \mathbb{E}[\log(1 - D_X(F(y)))]$
• G fools D_Y, F fools D_X
Cycle Consistency Loss (THE KEY INSIGHT)
G
F
$\approx x$
x
$G(x)$
$F(G(x))$
$\mathcal{L}_{\text{cyc}} = \mathbb{E}[\|F(G(x)) - x\|] + \mathbb{E}[\|G(F(y)) - y\|]$
• Forward cycle + Backward cycle
Full Objective
$\mathcal{L} = \mathcal{L}_{\text{GAN}}(G, D_Y) + \mathcal{L}_{\text{GAN}}(F, D_X) + \lambda \cdot \mathcal{L}_{\text{cyc}}$
• $\lambda = 10$ controls cycle consistency weight
G
F
Horse
Zebra
Horse (reconstructed, should match original)



Monet ⇄ Photos
Zebras ⇄ Horses
Summer ⇄ Winter
Monet → photo
zebra → horse
summer → winter
photo → Monet
horse → zebra
winter → summer
Photograph
Monet
Van Gogh
Cezanne
Ukiyo-e

See you on Monday!