

# Deep Generative Models

2021  
Multicampus

**Il Gu Yi**  
ModuLabs, Research Scientist  
**Soochul Park**  
GaudioLab, Research Scientist

# Disclaimer

- Details of convolutional neural networks
  - convolution, pooling operator, ...
- Details of recurrent neural networks
  - LSTM, GRU, ...
- Various regularization
  - Dropout, Batch normalization, ...
- Backpropagation



# Contents

- Introduction to Generative Models
- Review of Probability Theory
- Autoregressive Models
- Latent Variable Models
- Flow based Models
- Generative Adversarial Networks

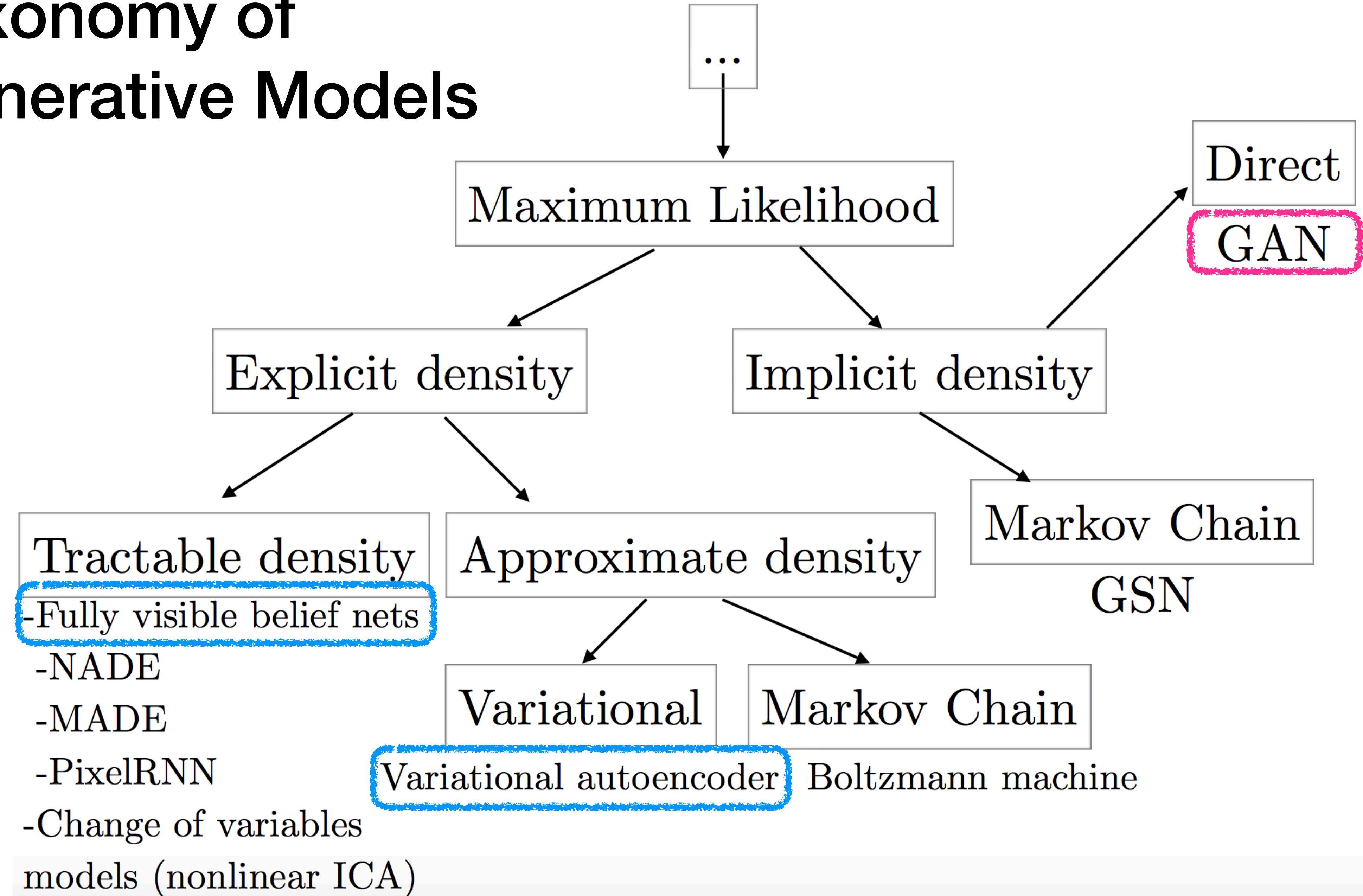


# References

- Deep Learning Book, Ian Goodfellow, et. al., 2015
- Stanford cs236 fall18: Deep Generative Models
- A lot of papers and codes

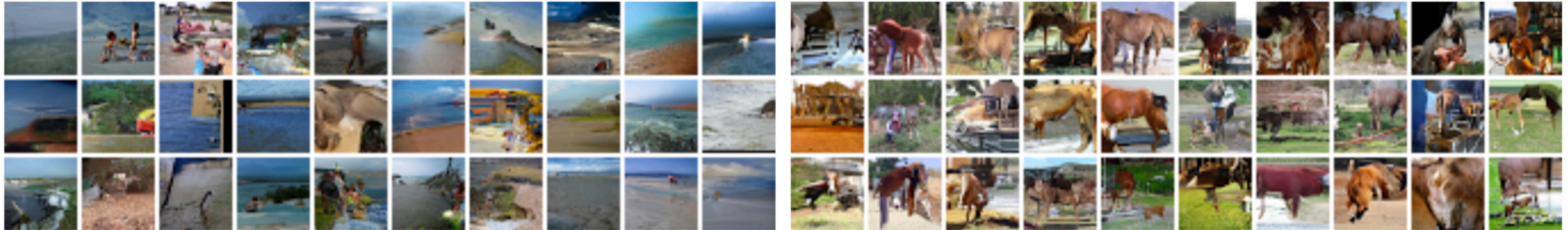


# Taxonomy of Generative Models

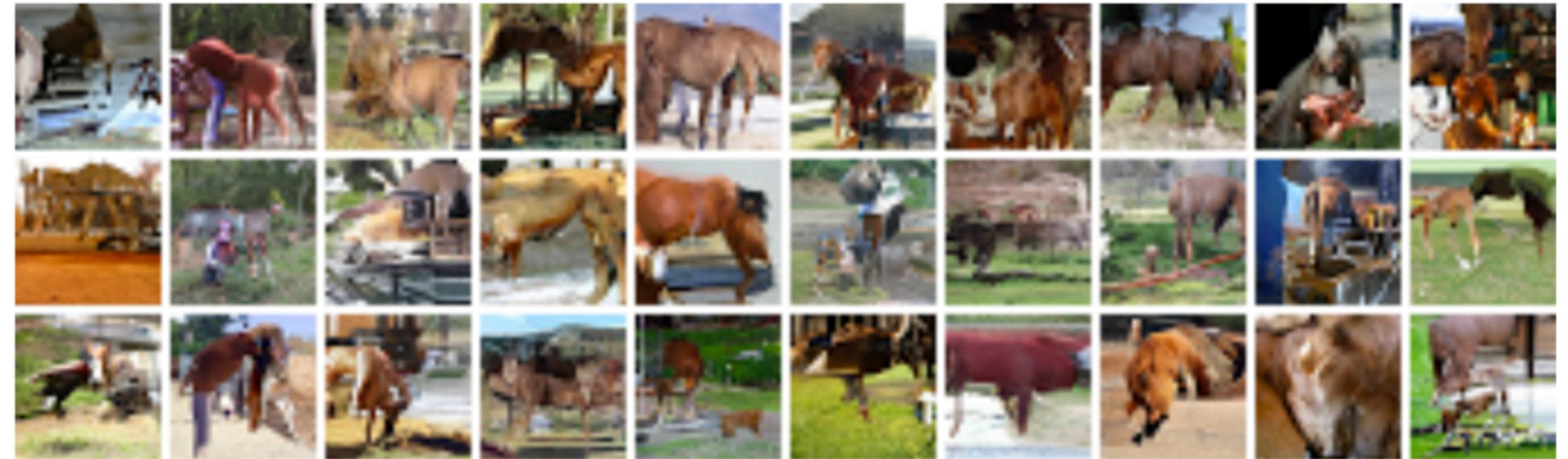


# AutoRegressive Models

# PixelCNN



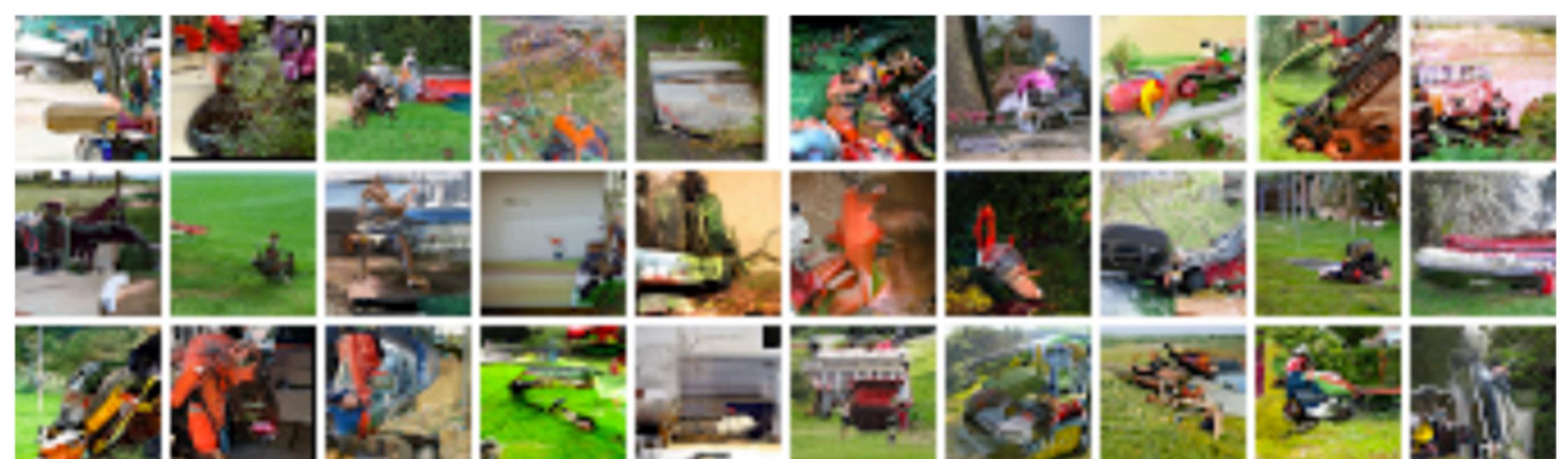
Sandbar



Sorrel horse

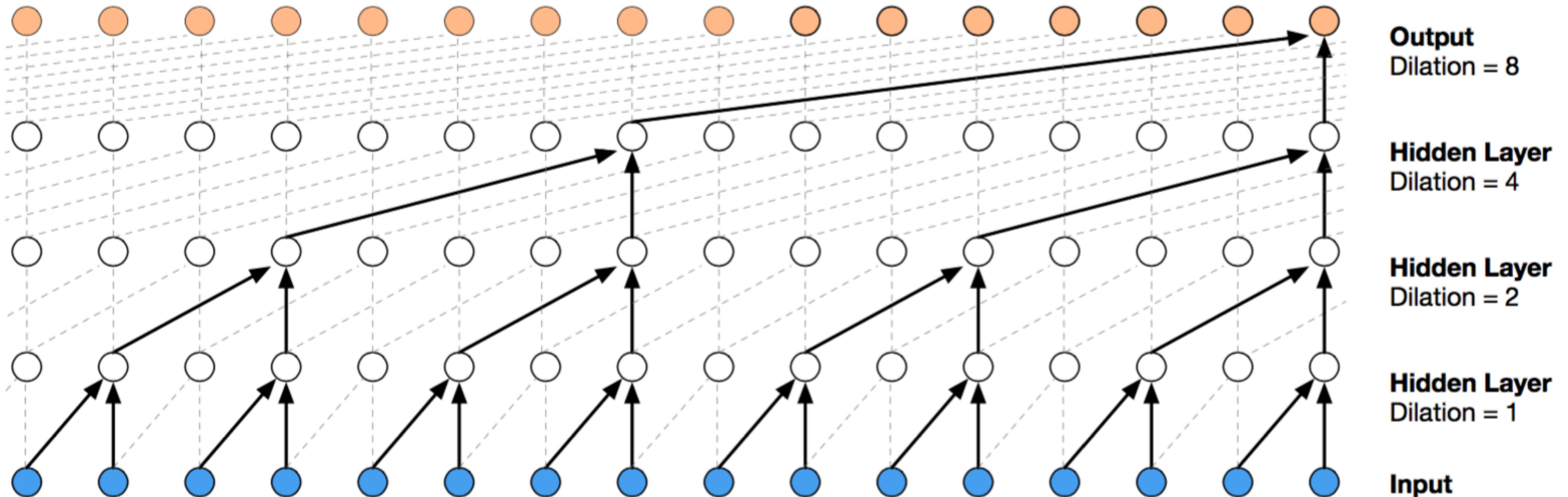


Lhasa Apso (dog)



Lawn mower

# WaveNet



**Audio samples**

<https://google.github.io/tacotron/publications/tacotron2/index.html>



# Transformer

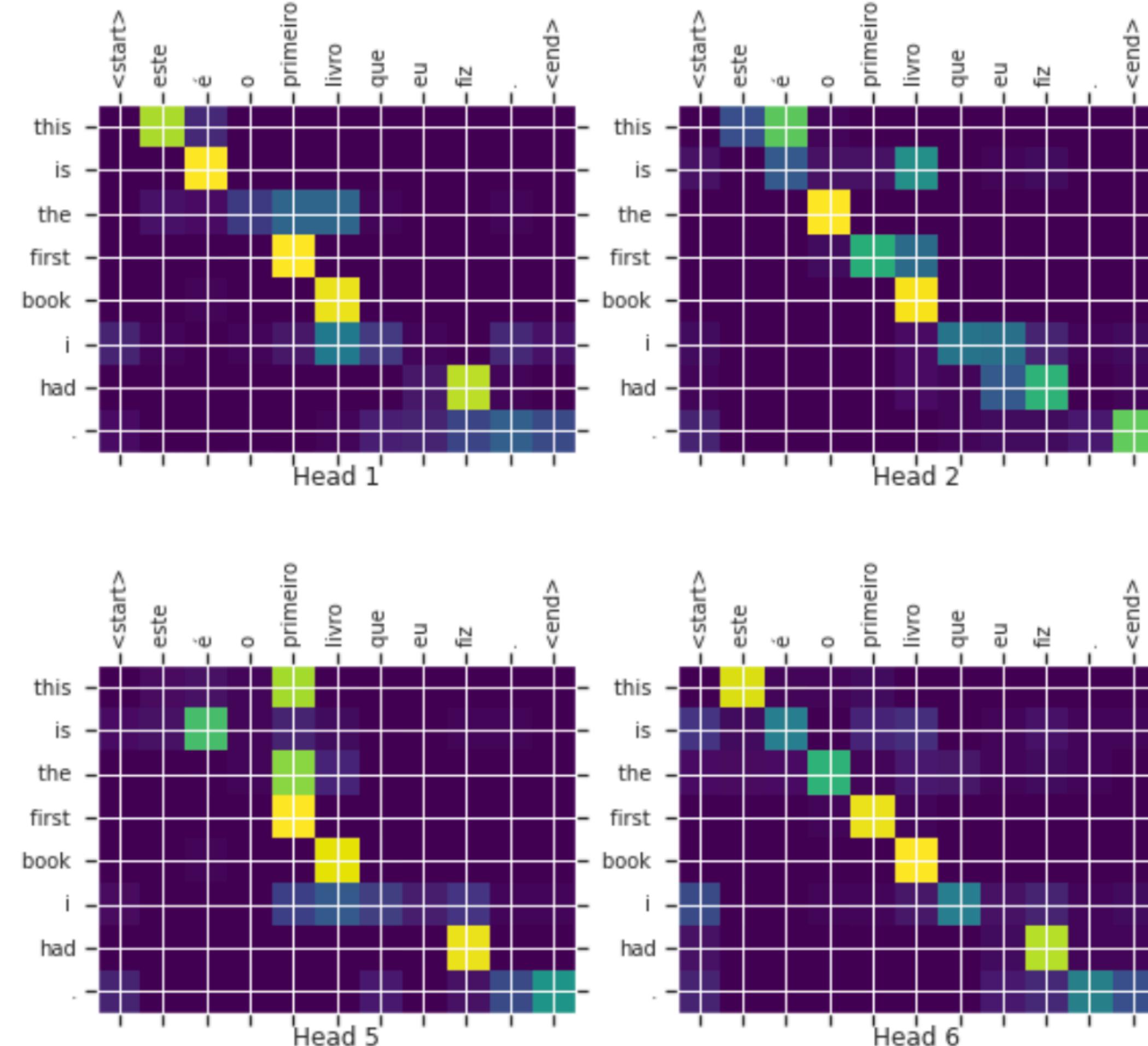
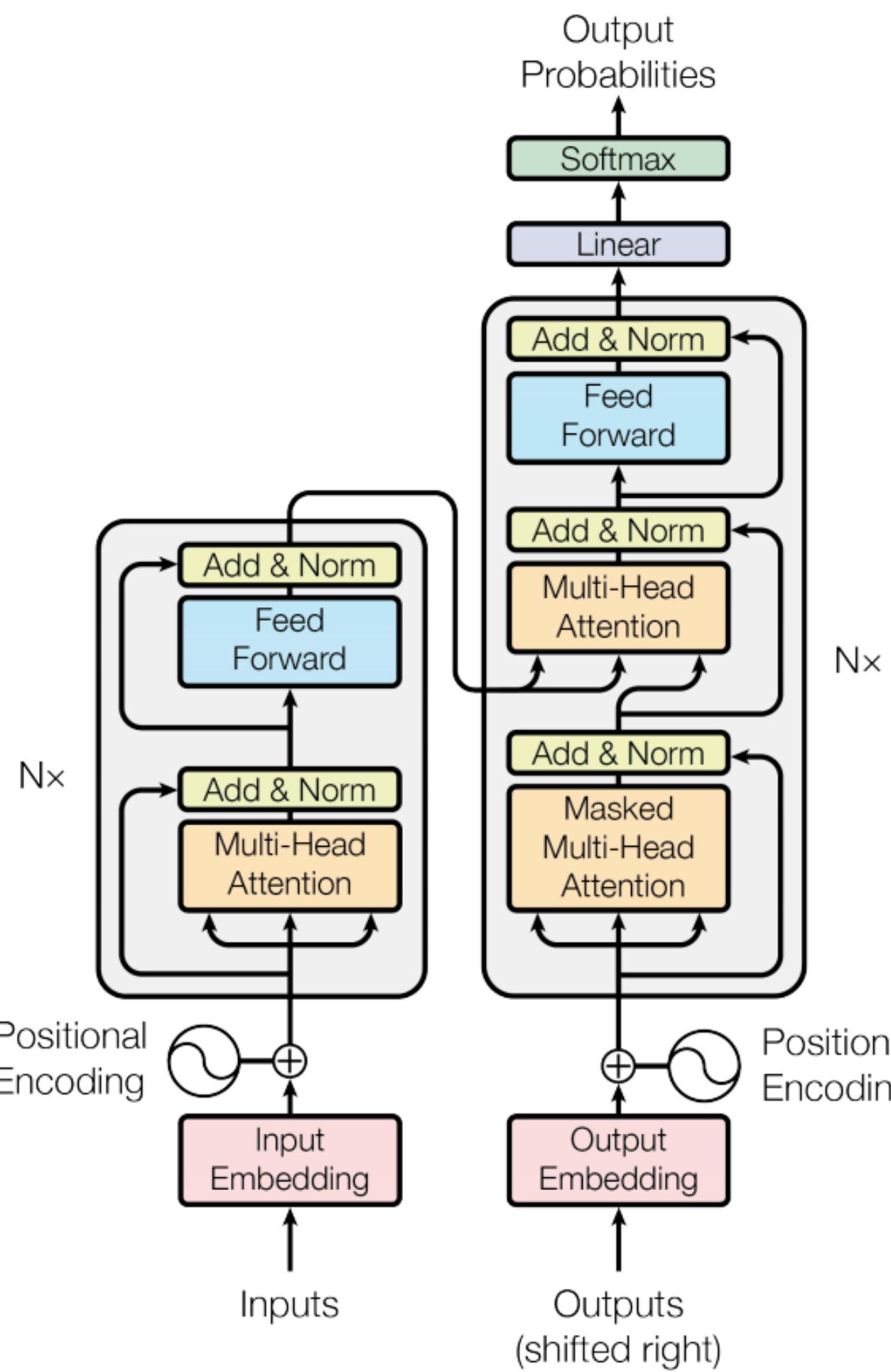
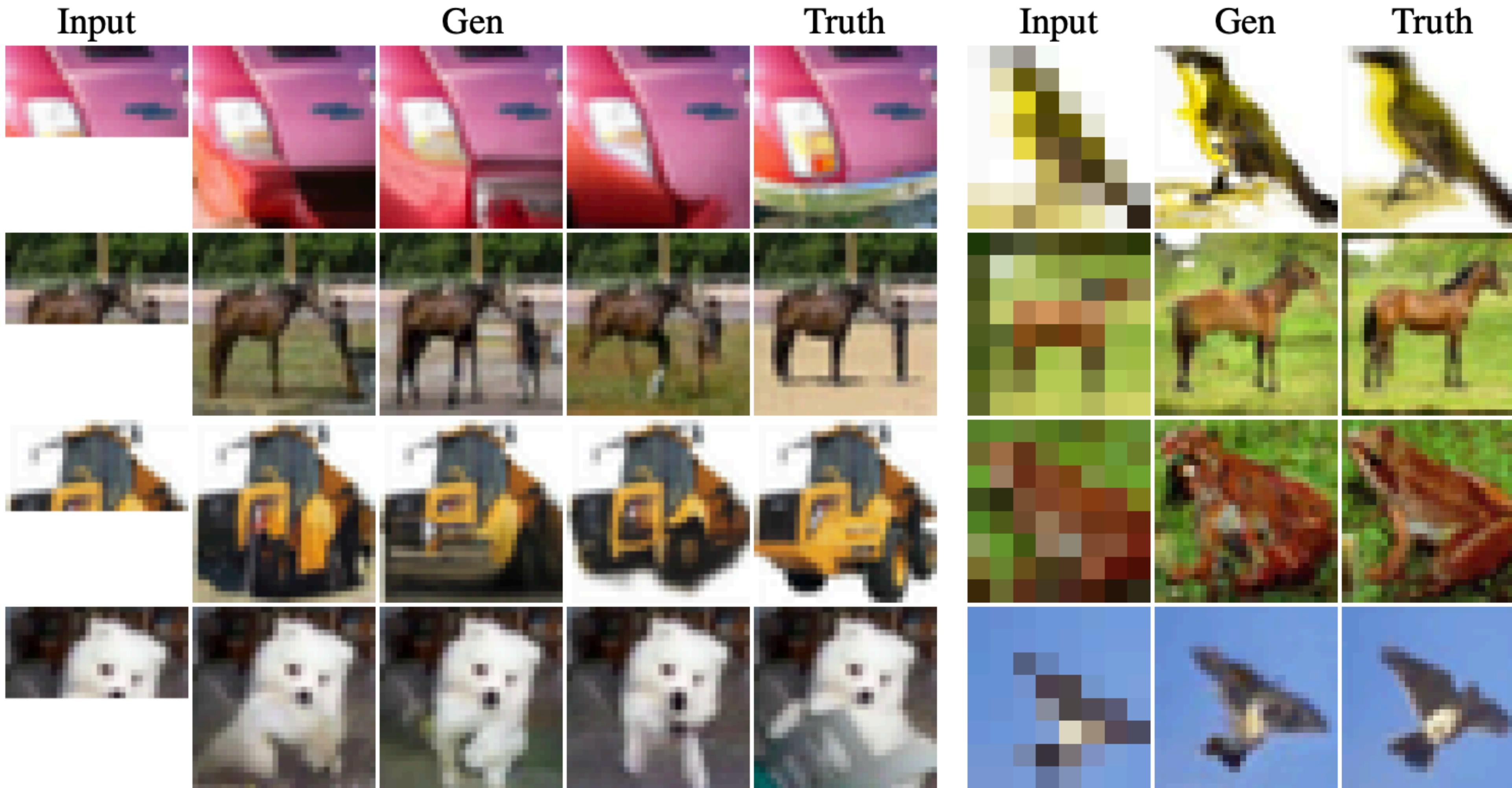


Figure credit :

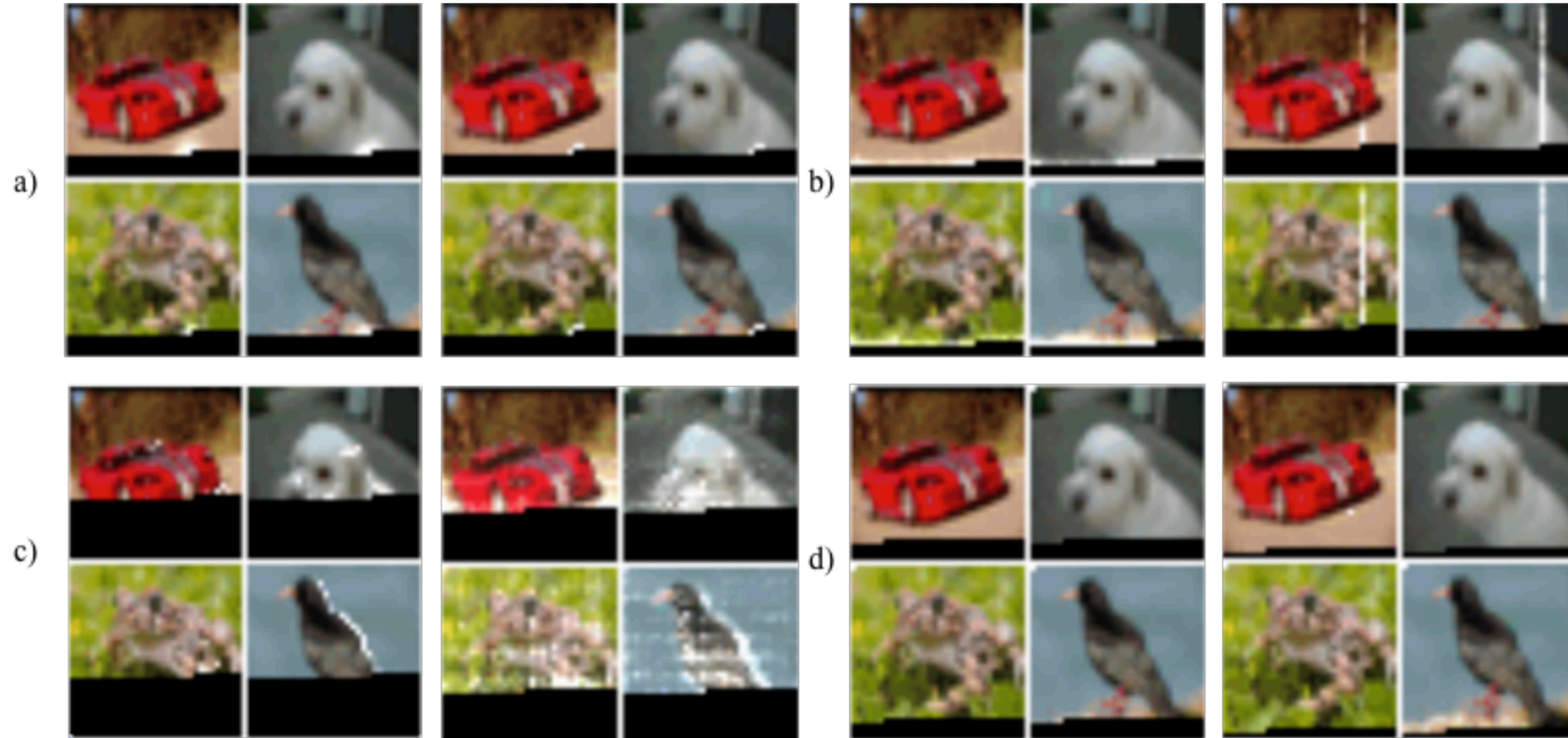
Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems. 2017.

<https://www.tensorflow.org/text/tutorials/transformer>

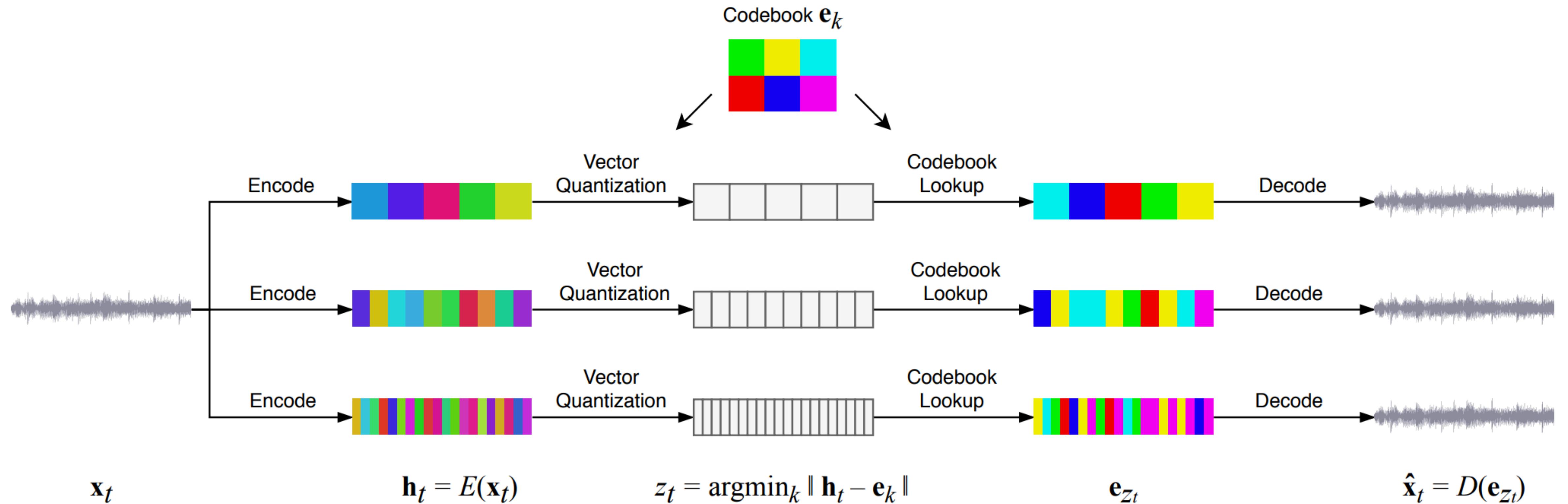
# Image Transformer



# Sparse Transformer



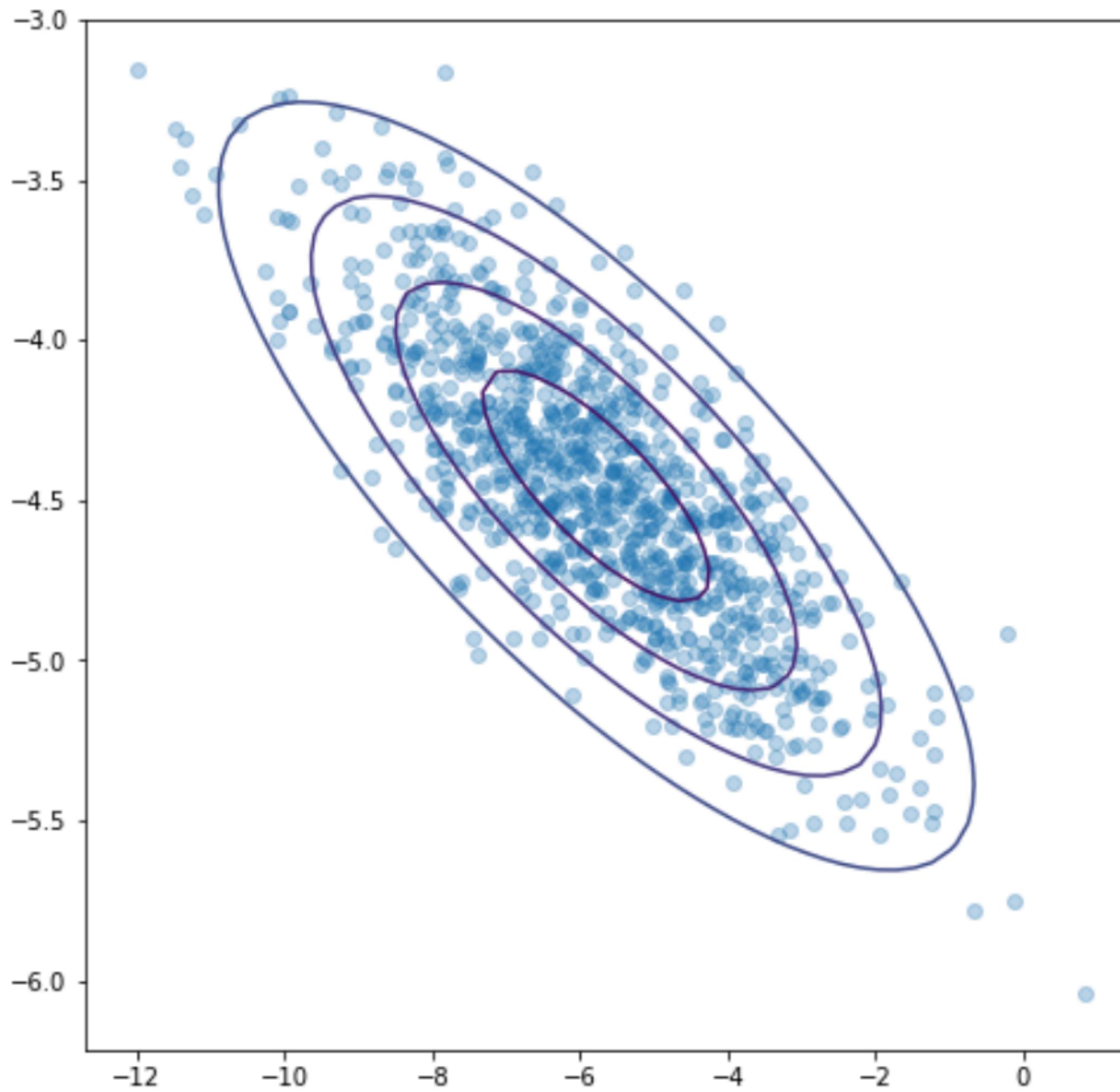
# Jukebox



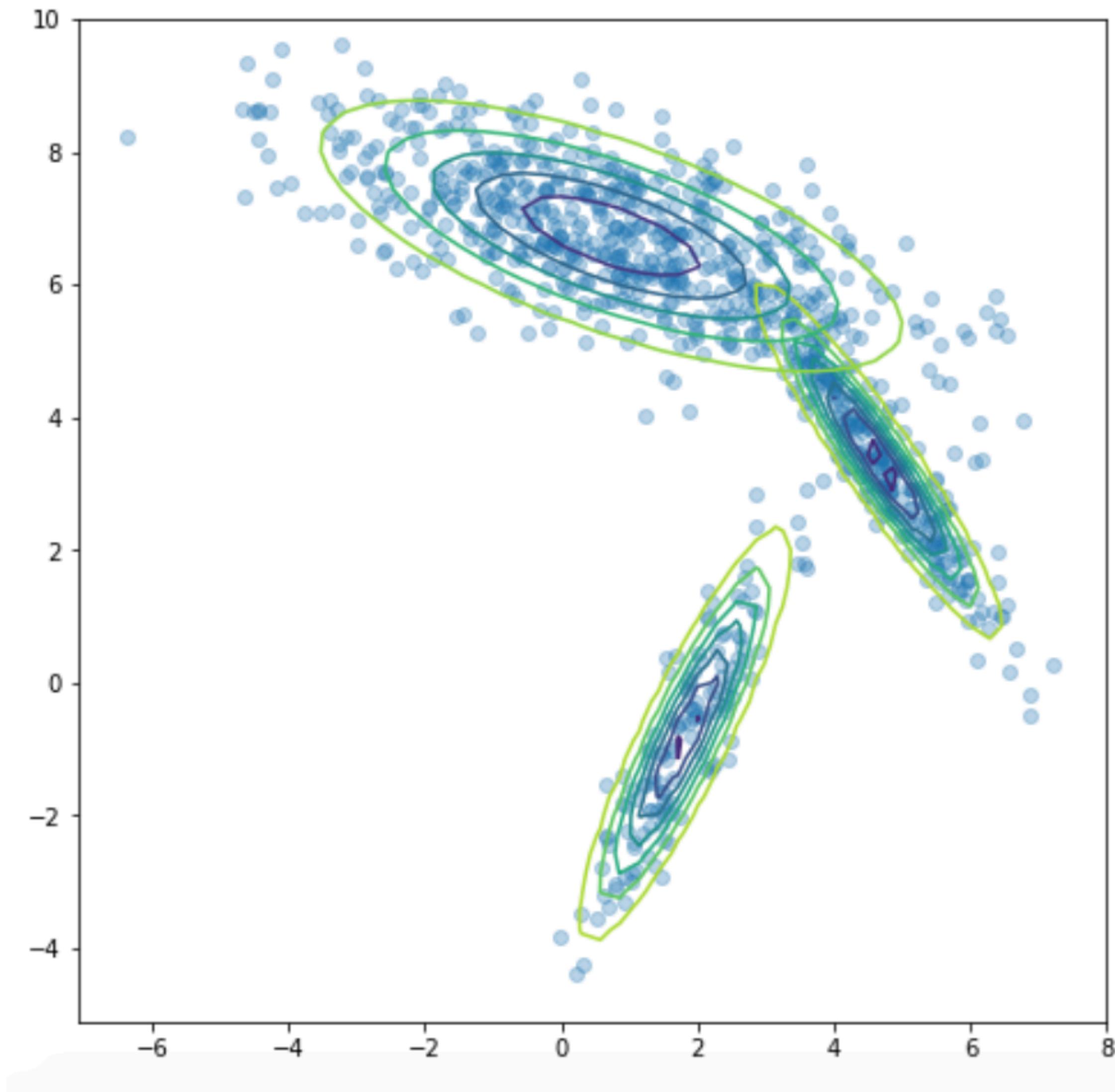
$$\begin{aligned}
 p(\mathbf{z}) &= p(\mathbf{z}^{\text{top}}, \mathbf{z}^{\text{middle}}, \mathbf{z}^{\text{bottom}}) \\
 &= p(\mathbf{z}^{\text{top}})p(\mathbf{z}^{\text{middle}}|\mathbf{z}^{\text{top}})p(\mathbf{z}^{\text{bottom}}|\mathbf{z}^{\text{middle}}, \mathbf{z}^{\text{top}})
 \end{aligned}$$

# Latent Models

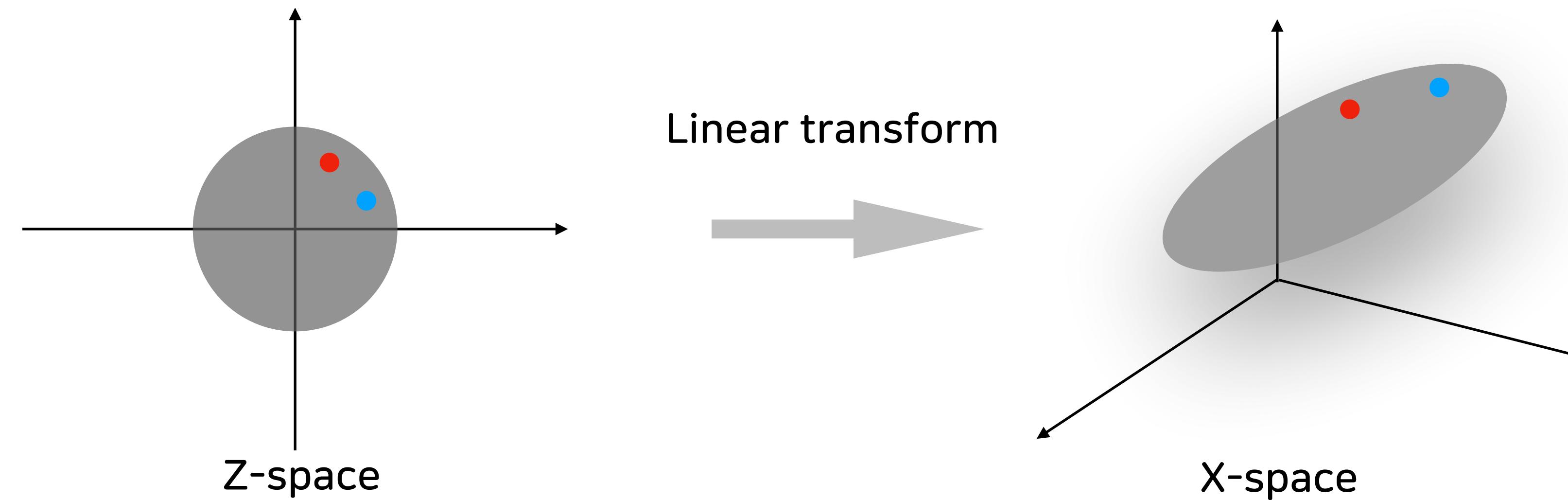
# Gaussian Model



# Gaussian Mixture Models



# Probabilistic PCA



Prior  $p_\theta(z) = N(z | 0, I)$

Conditional  $p_\theta(x | z) = N(x | Wz + \mu, \sigma^2 I)$

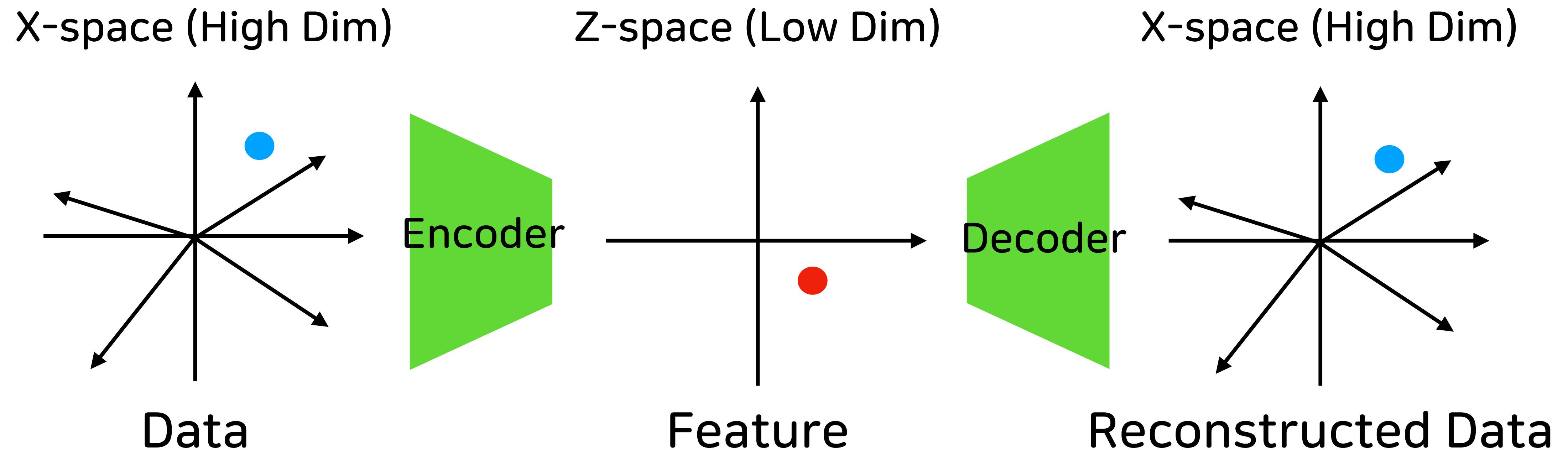
(Linear-Gaussian Model)

References :

Michael E. Tipping and Christopher M. Bishop. Probabilistic Principal Component Analysis. 1999.

Christopher M. Bishop. Pattern Recognition and Machine Learning. p.570-580

# Auto-Encoder



# Variational Auto-Encoder

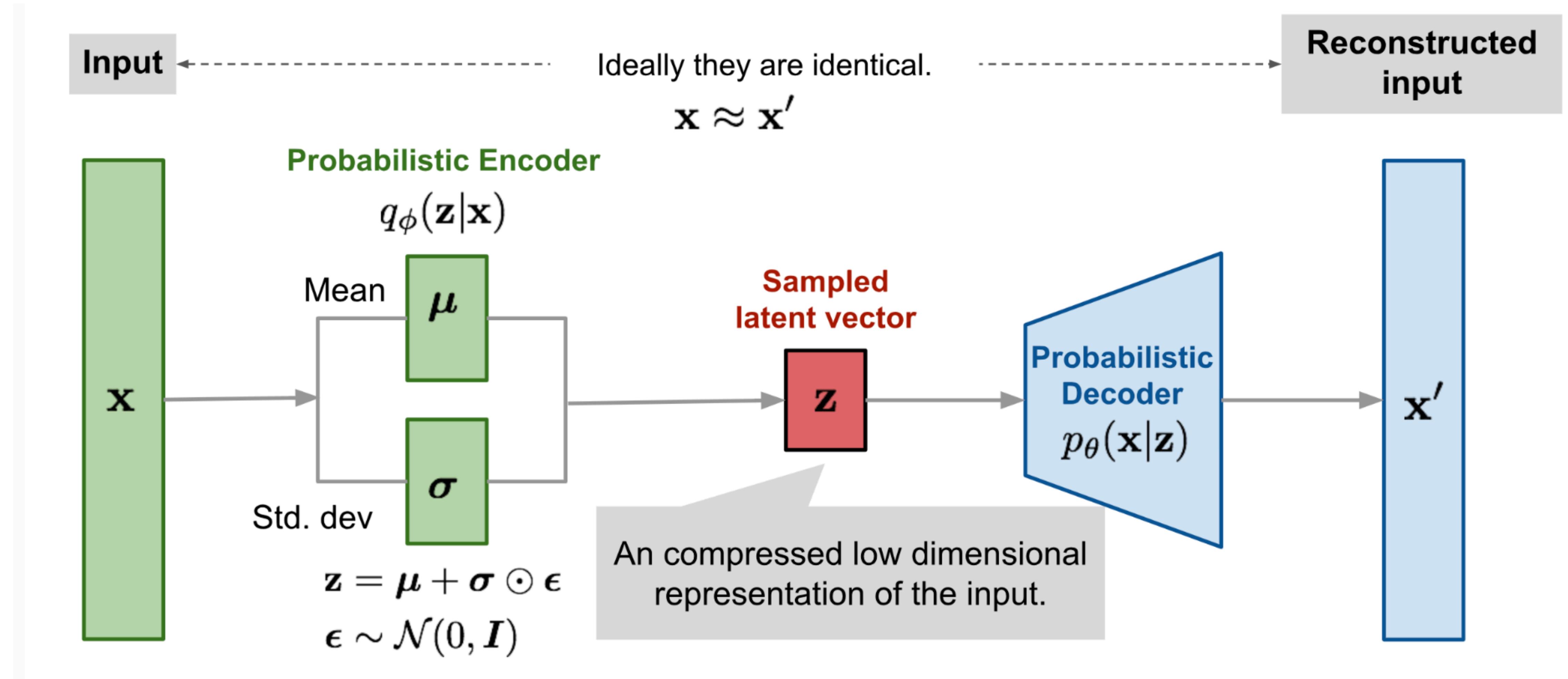
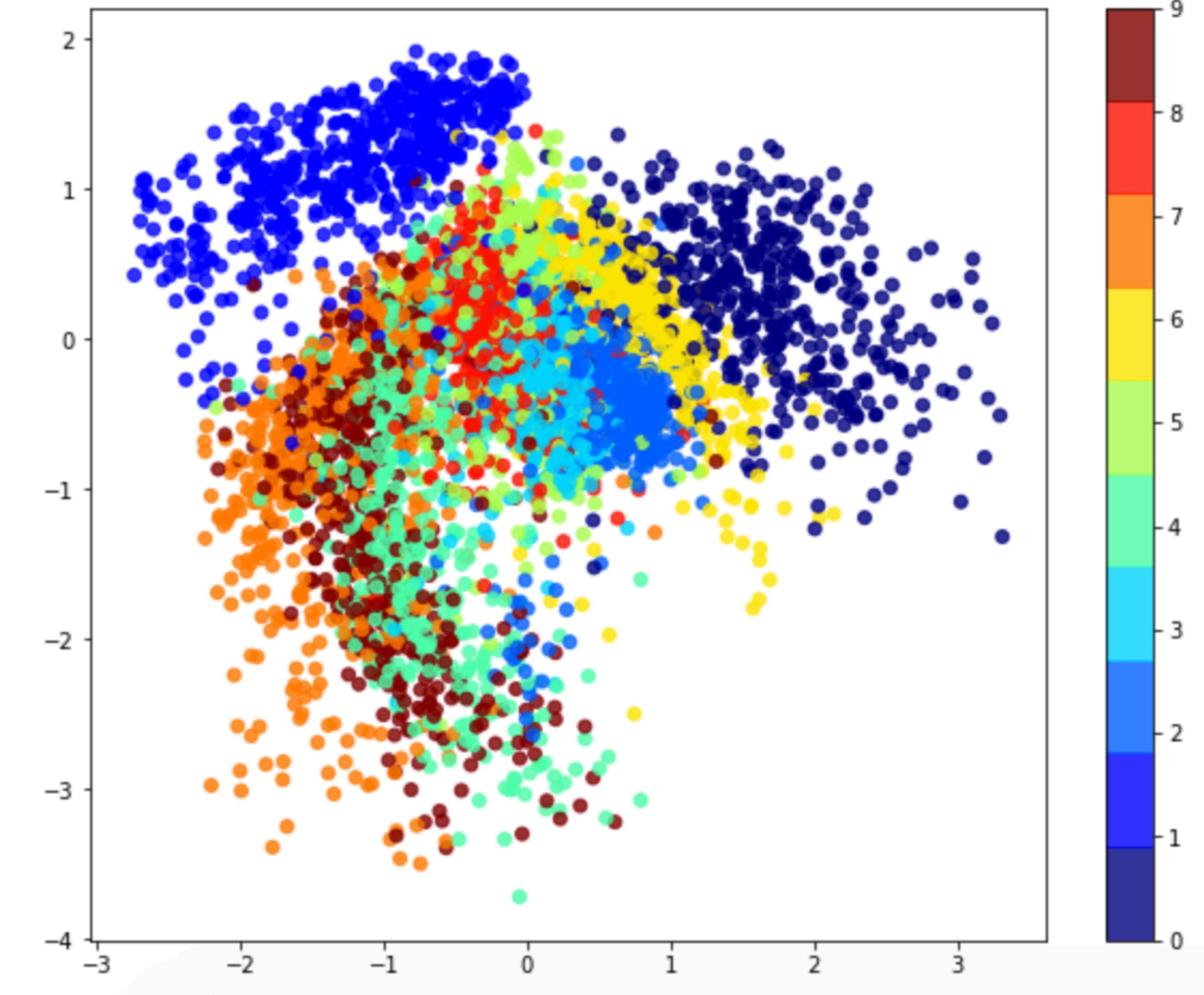
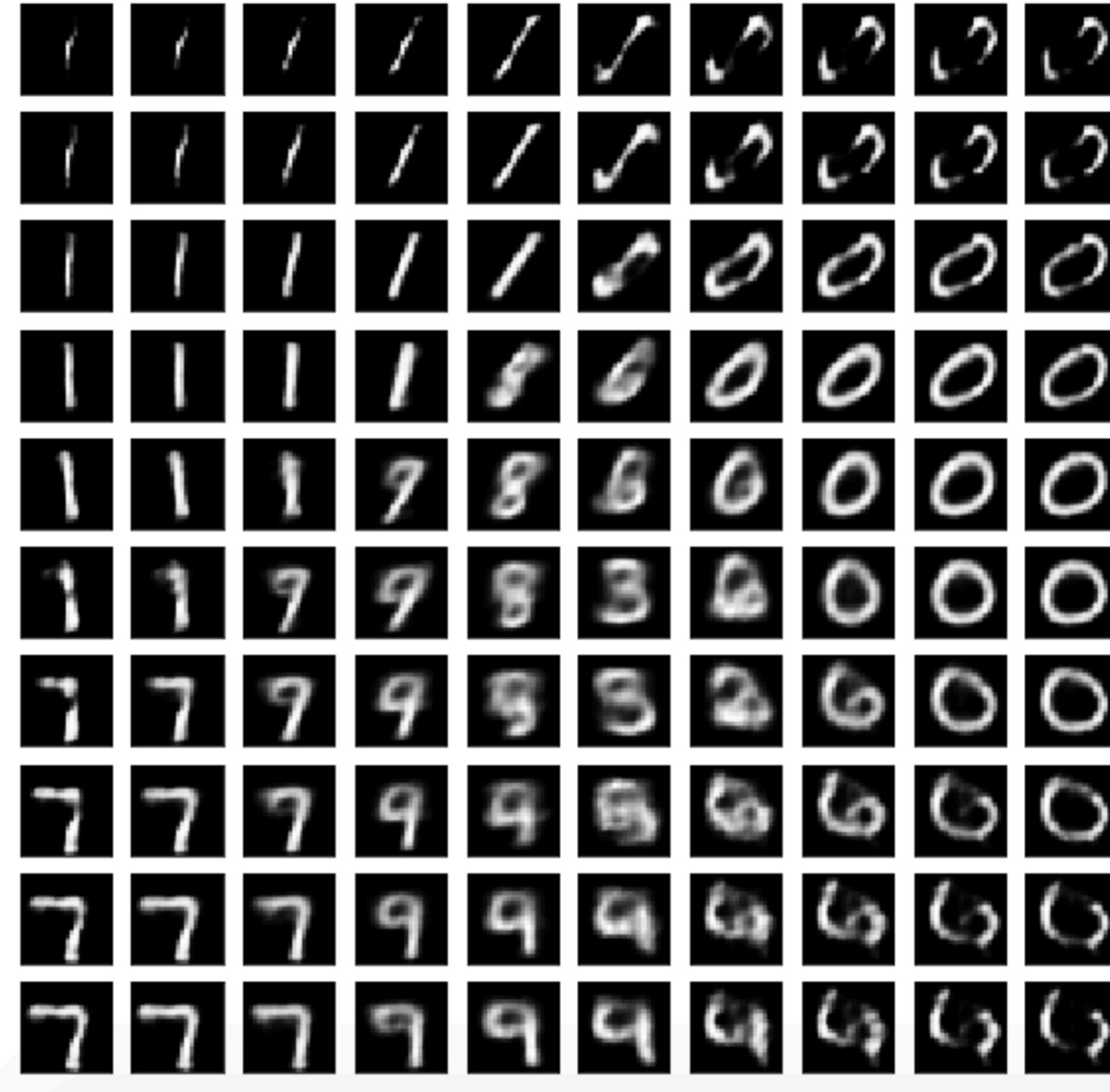


Fig. 9. Illustration of variational autoencoder model with the multivariate Gaussian assumption.

# Variational Auto-Encoders

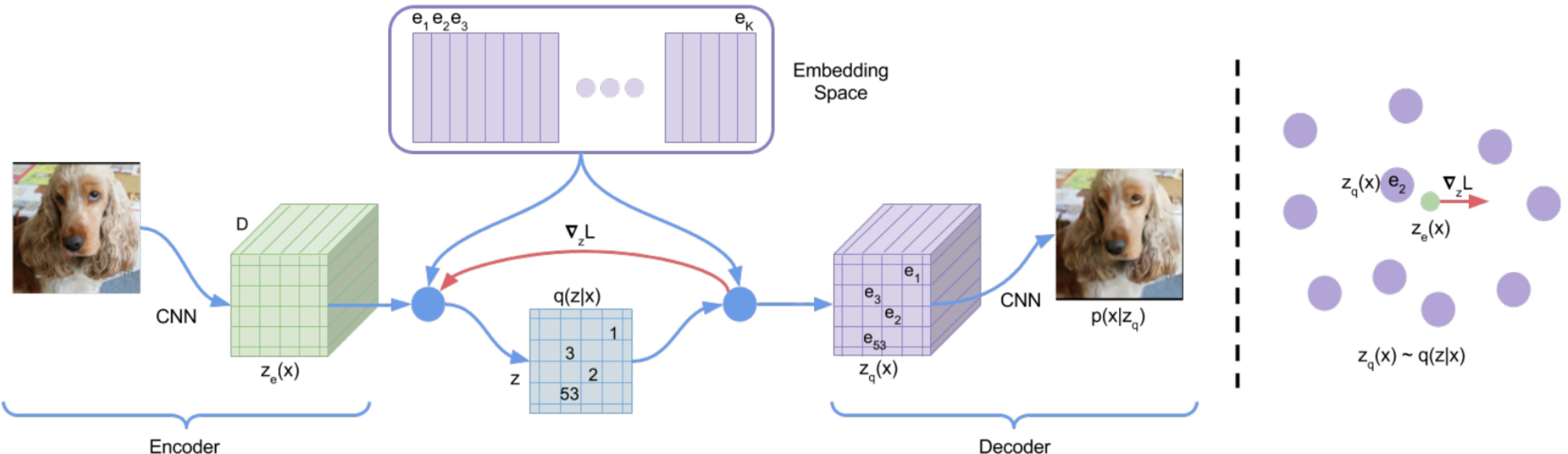


# Hierachical VAE



Child, Rewon. "Very deep vaes generalize autoregressive models and can outperform them on images." arXiv preprint arXiv:2011.10650 (2020).

# VQ-VAE



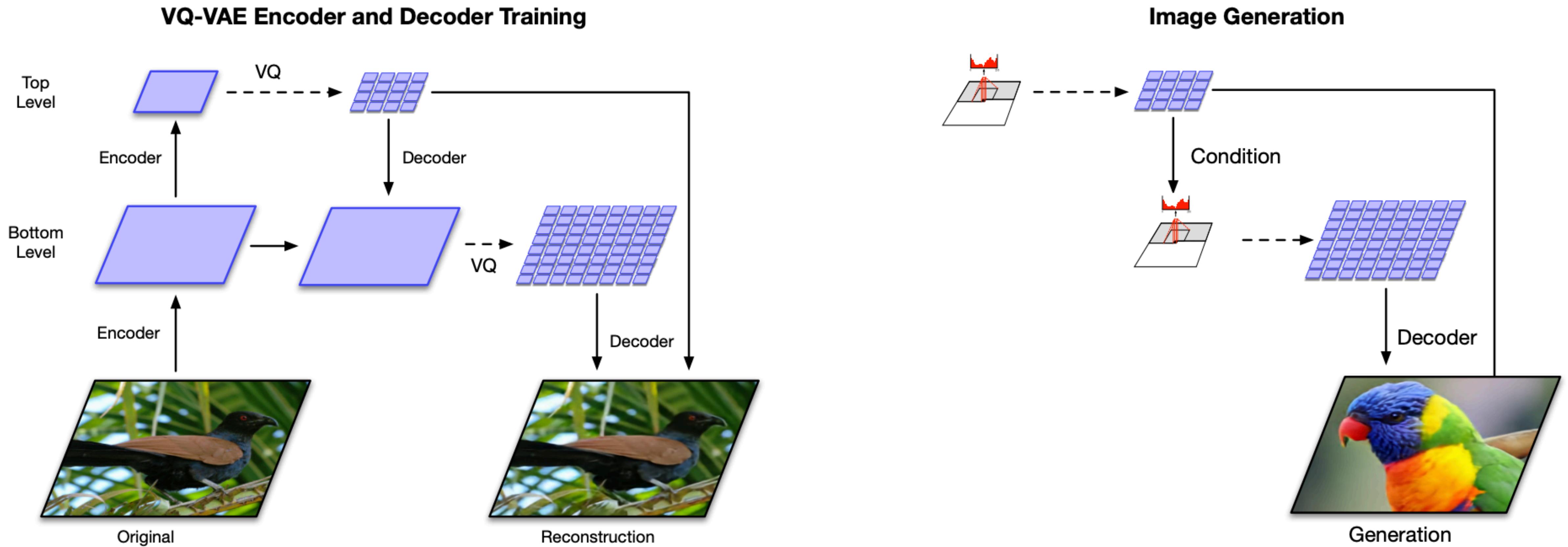
$$\mathcal{L}(\mathbf{x}, D(\mathbf{e})) = \|\mathbf{x} - D(\mathbf{e})\|_2^2 + \|sg[E(\mathbf{x})] - \mathbf{e}\|_2^2 + \beta \|sg[\mathbf{e}] - E(\mathbf{x})\|_2^2$$

**Reconstruction loss**

**Codebook loss**

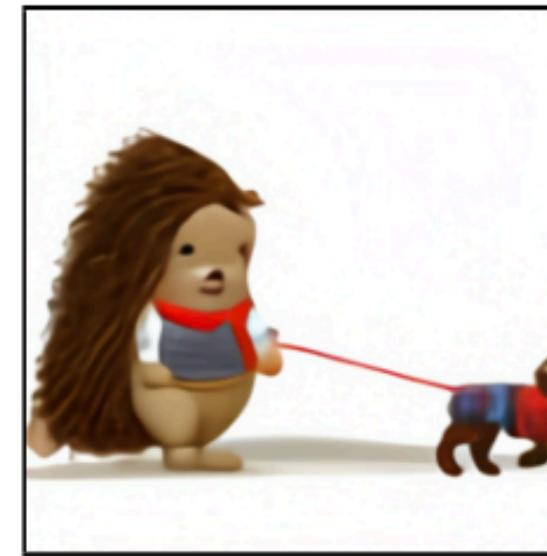
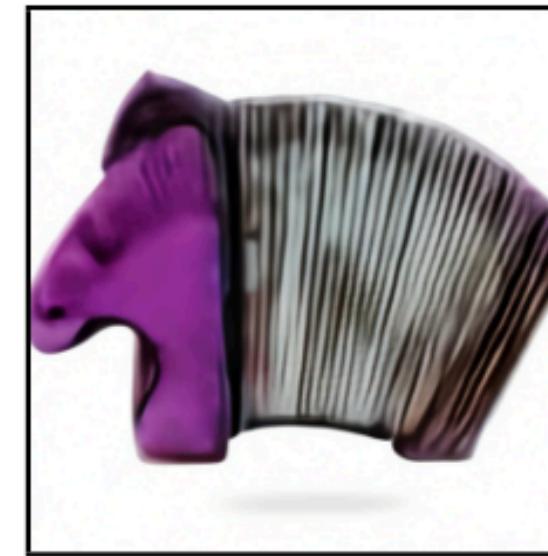
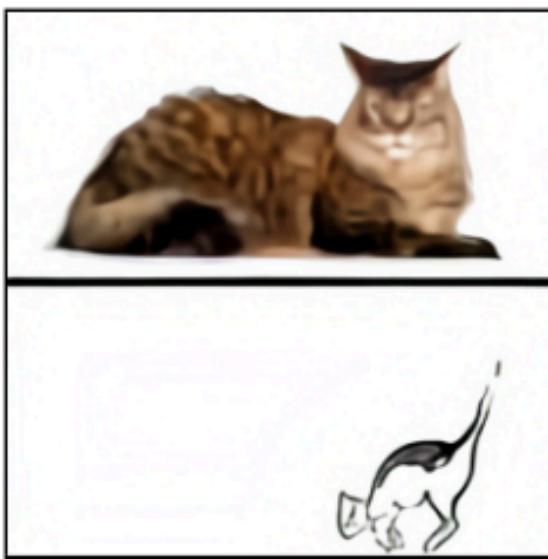
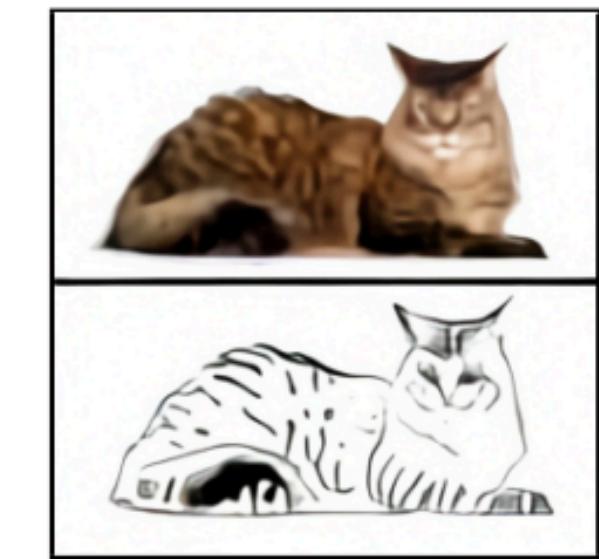
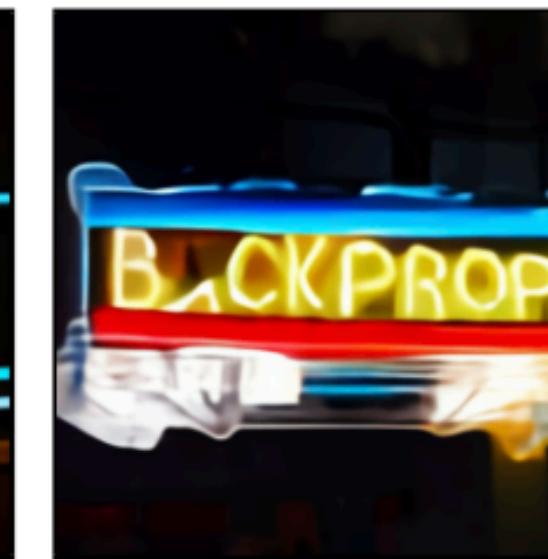
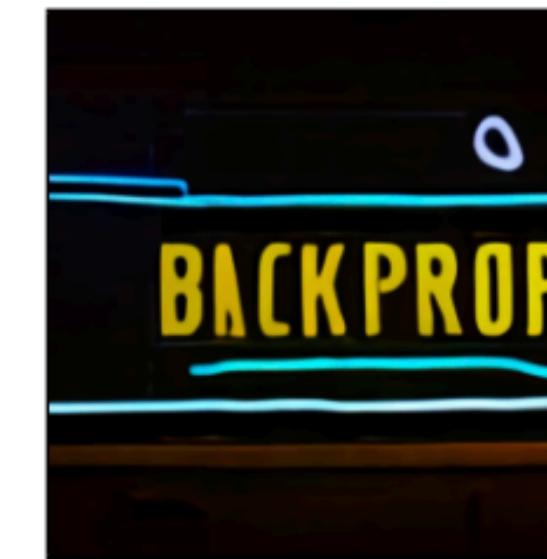
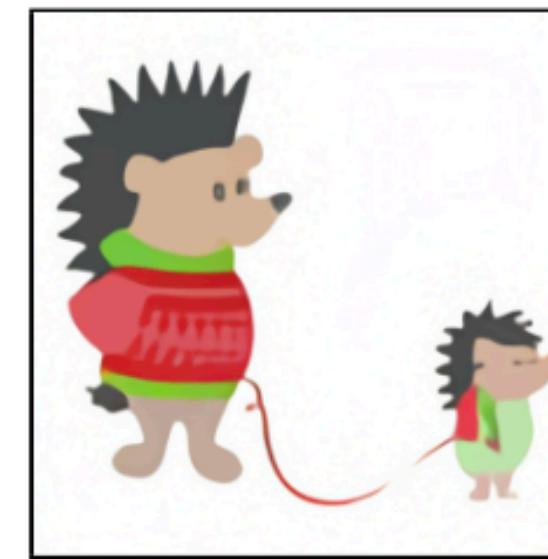
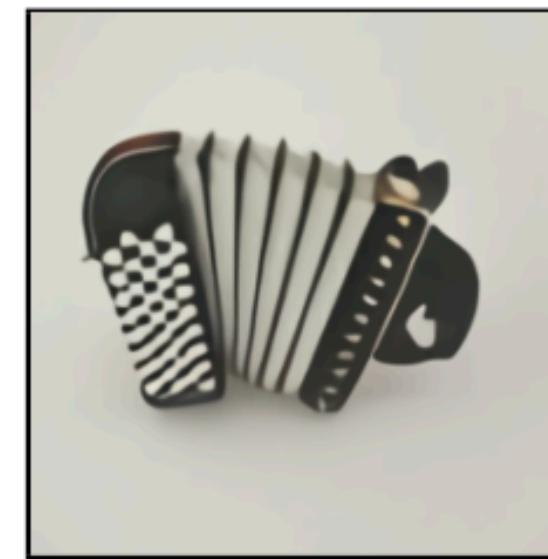
**Commitment loss**

# VQ-VAE 2



Razavi, Ali, Aaron van den Oord, and Oriol Vinyals. "Generating diverse high-fidelity images with vq-vae-2." *Advances in neural information processing systems*. 2019.

# DALL·E



(a) a tapir made of accordion.  
a tapir with the texture of an  
accordion.

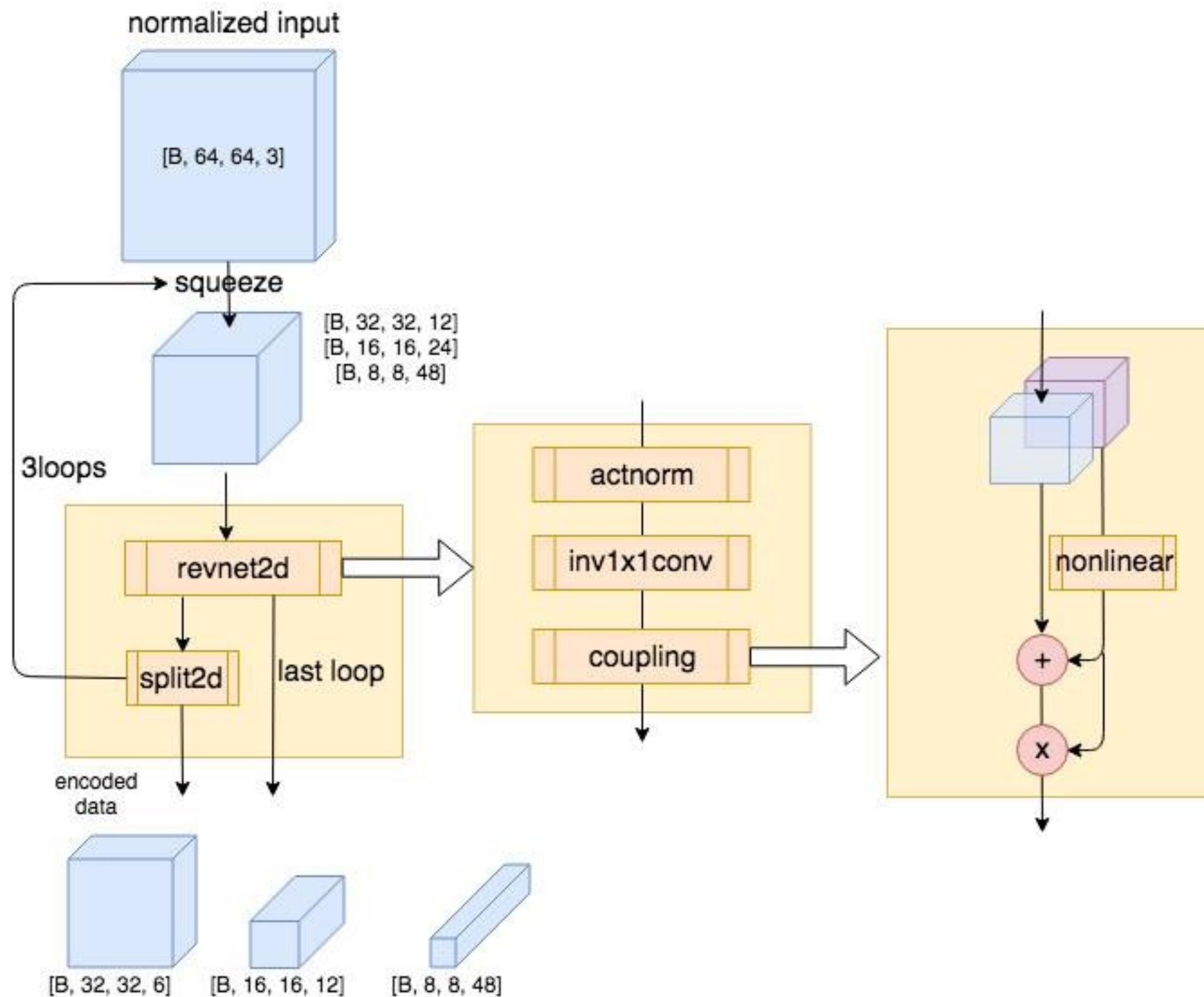
(b) an illustration of a baby  
hedgehog in a christmas  
sweater walking a dog

(c) a neon sign that reads  
“backprop”. a neon sign that  
reads “backprop”. backprop  
neon sign

(d) the exact same cat on the  
top as a sketch on the bottom

# Flow Models

# GLOW networks



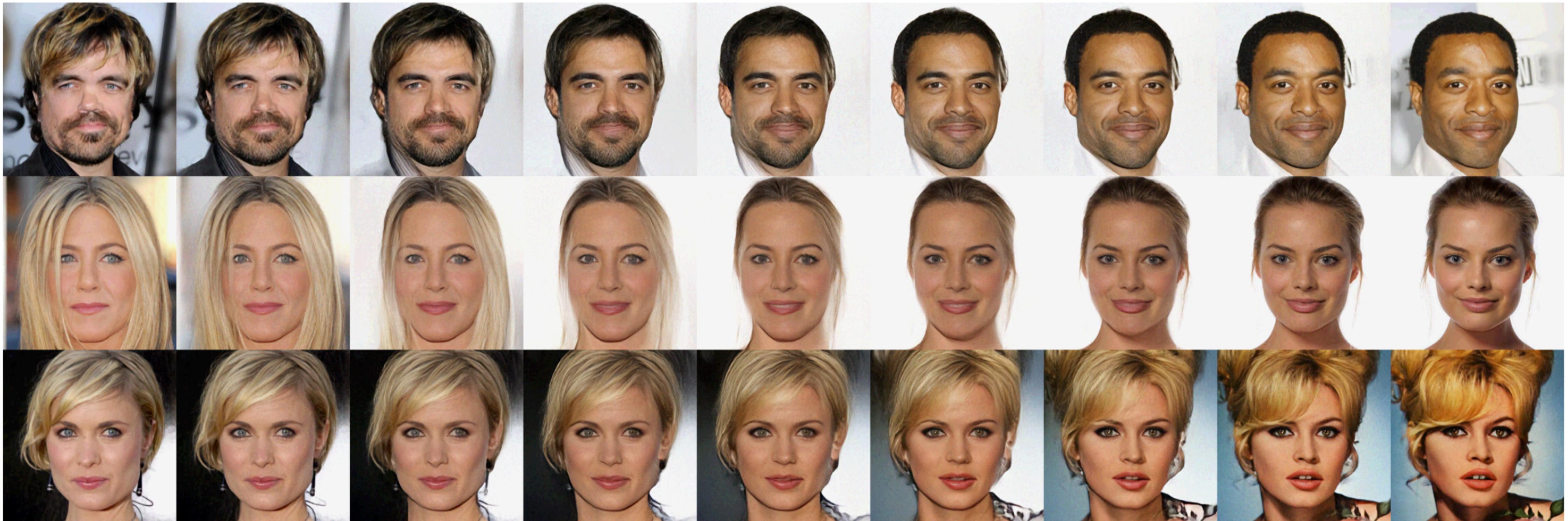
# Results via NICE



# Interpolations via RealNVP



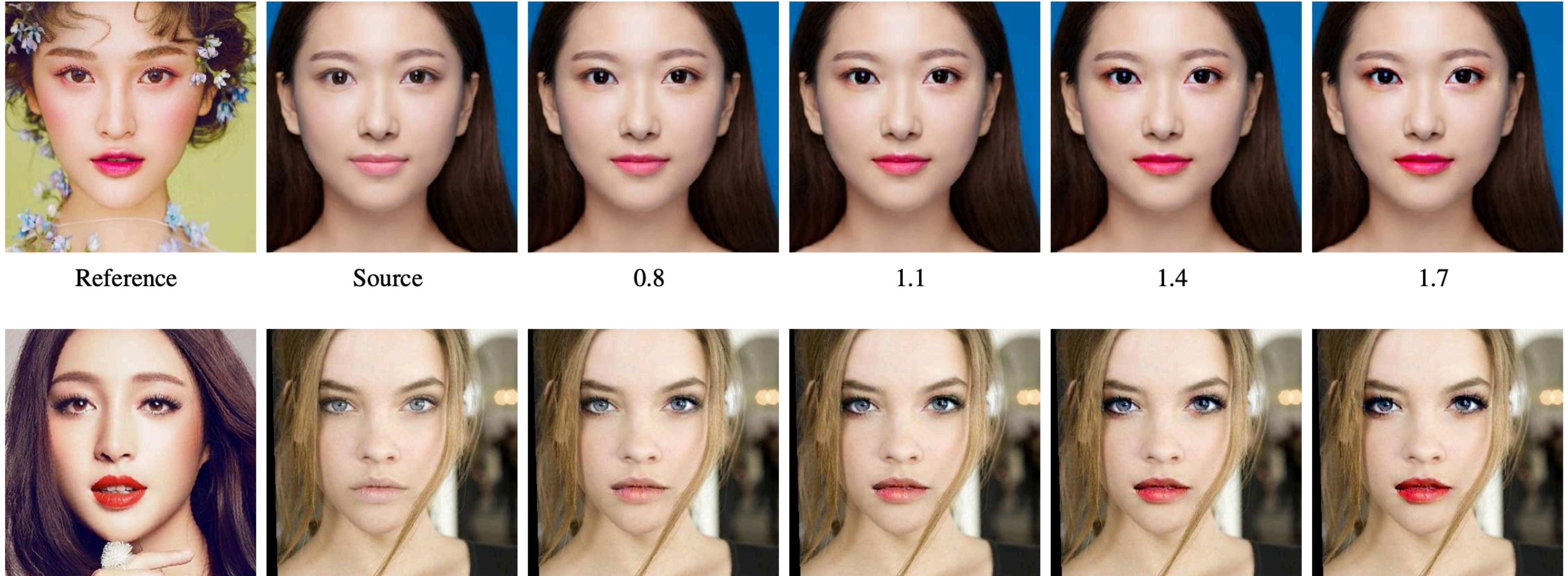
# Interpolations via Glow



Demo: <https://openai.com/blog/glow/>

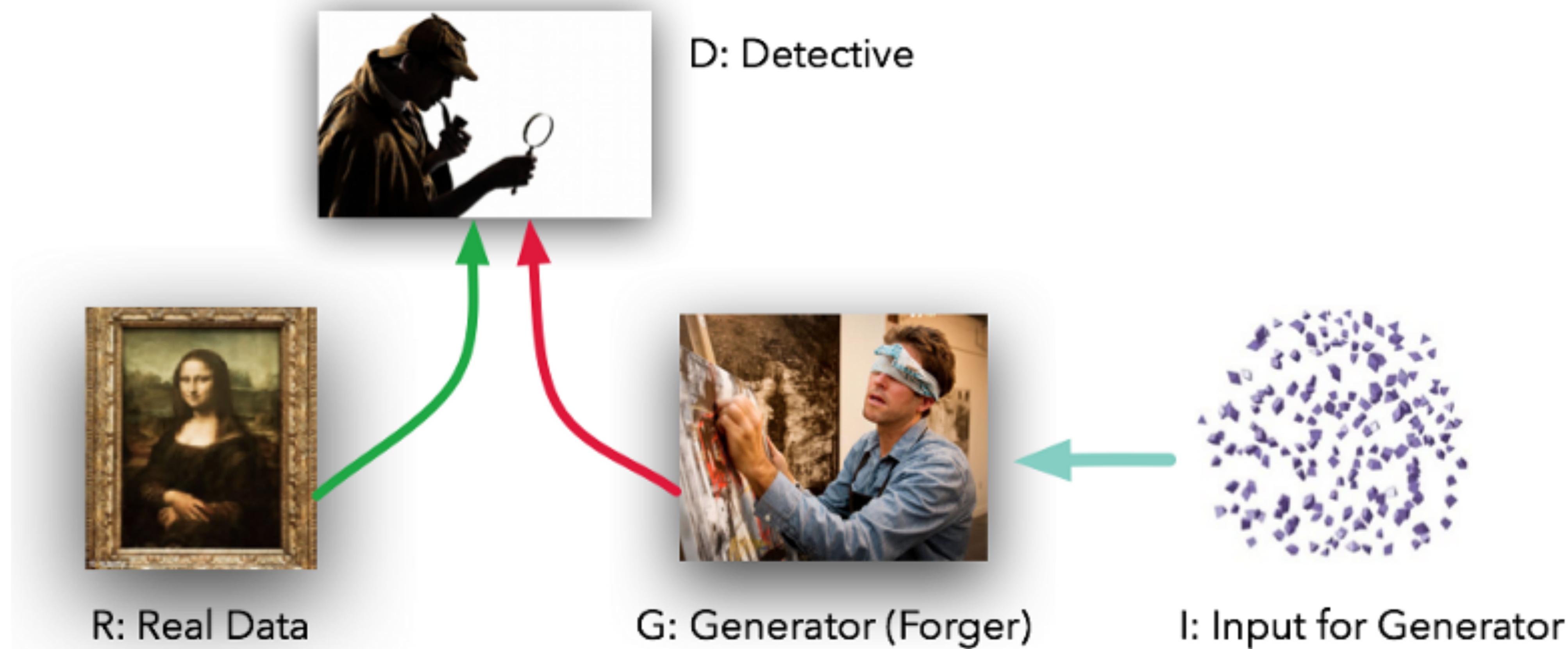
Figure credit: D. Kingma, et. al., Glow: Generative Flow with Invertible 1x1 Convolutions

# BeautyGLOW

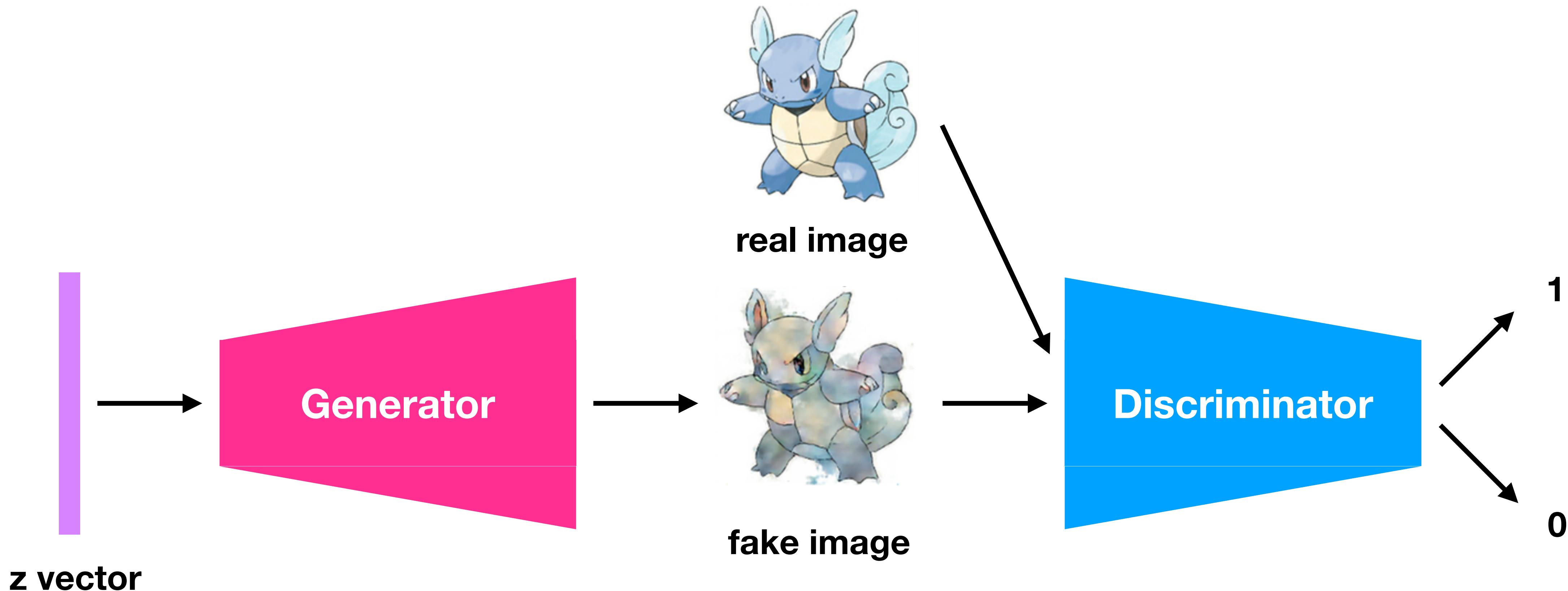


# Generative Adversarial Networks

# Generative Adversarial Networks Framework

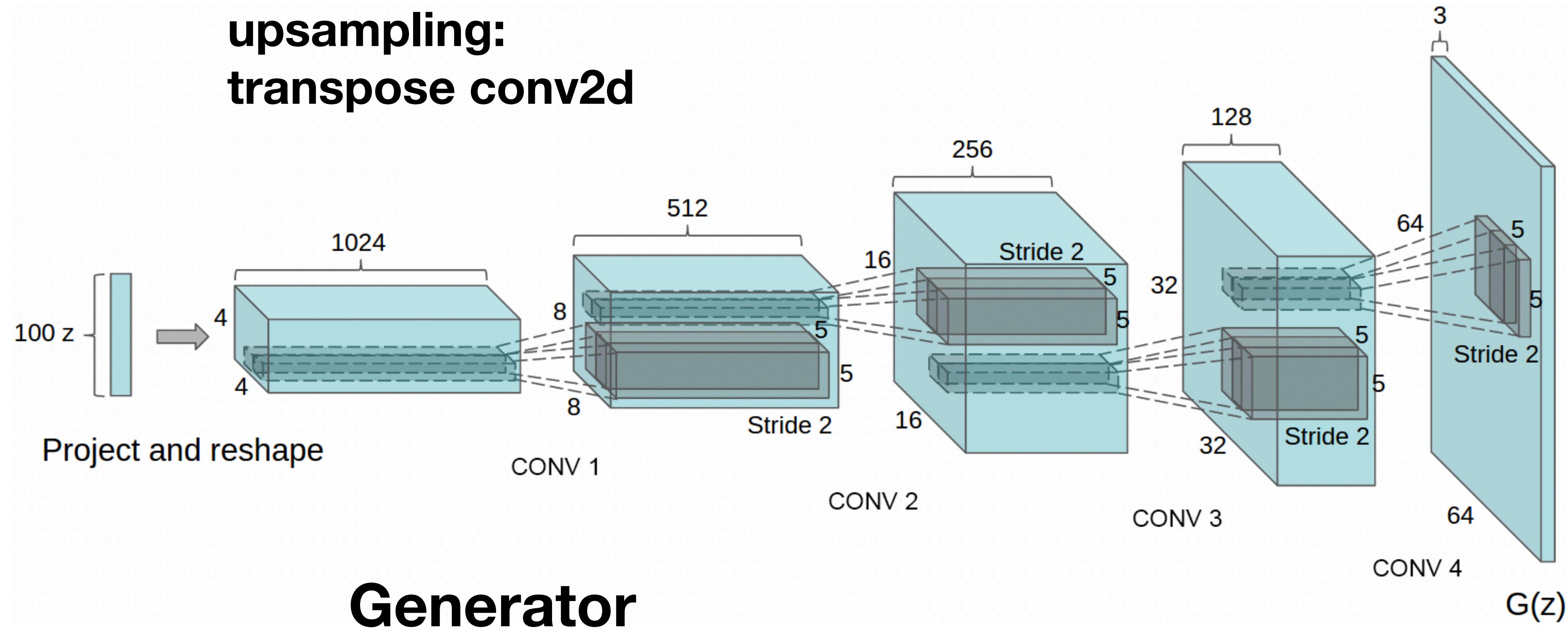


# Generative Adversarial Networks Framework



# DCGAN Architectures

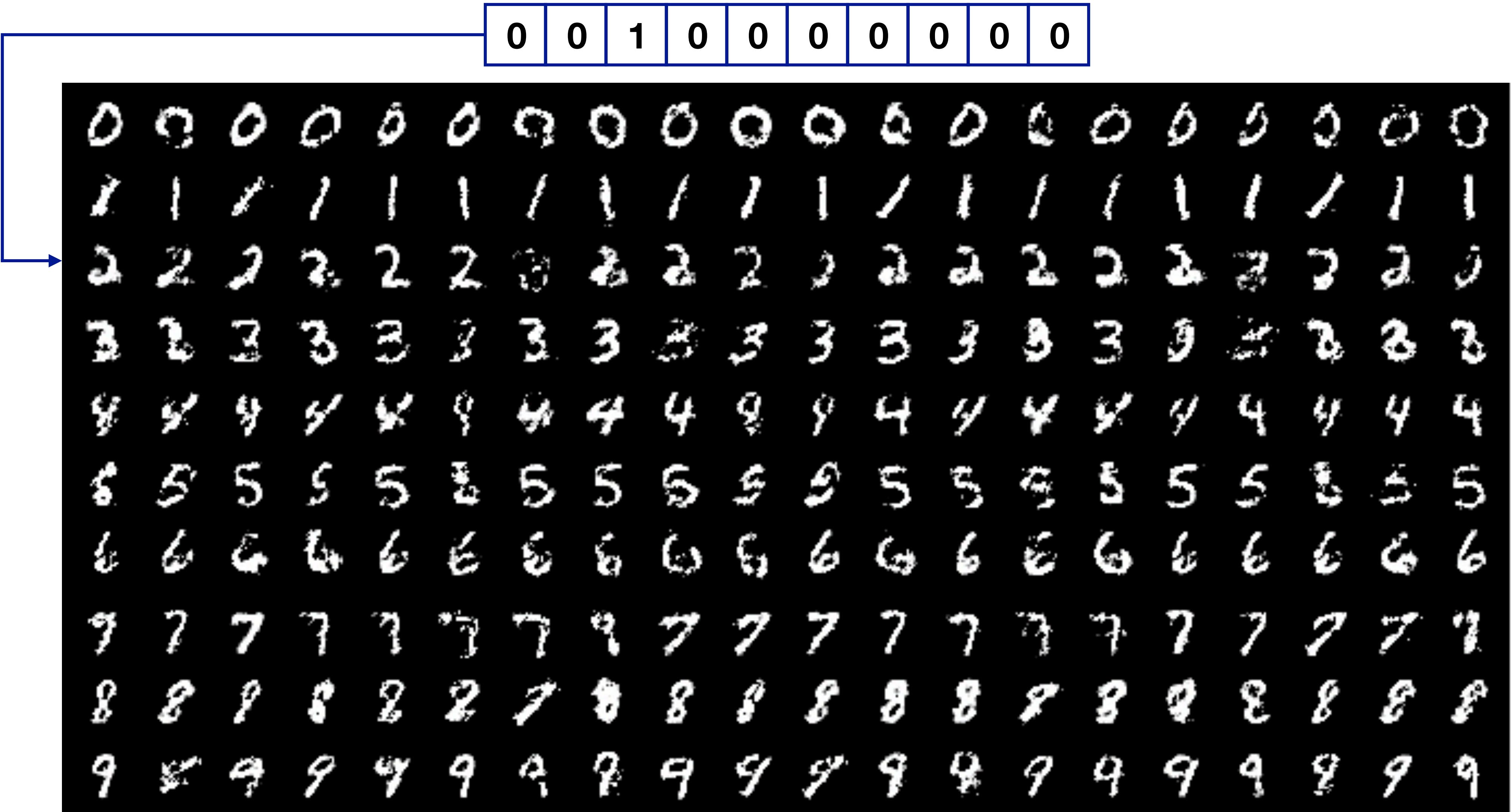
**upsampling:**  
**transpose conv2d**



# DCGAN Results



# Conditional GAN Results



# InfoGAN Results



(a) Varying  $c_1$  on InfoGAN (Digit type)



(b) Varying  $c_1$  on regular GAN (No clear meaning)



(c) Varying  $c_2$  from  $-2$  to  $2$  on InfoGAN (Rotation)



(d) Varying  $c_3$  from  $-2$  to  $2$  on InfoGAN (Width)

# InfoGAN Results



(a) Azimuth (pose)

(b) Elevation

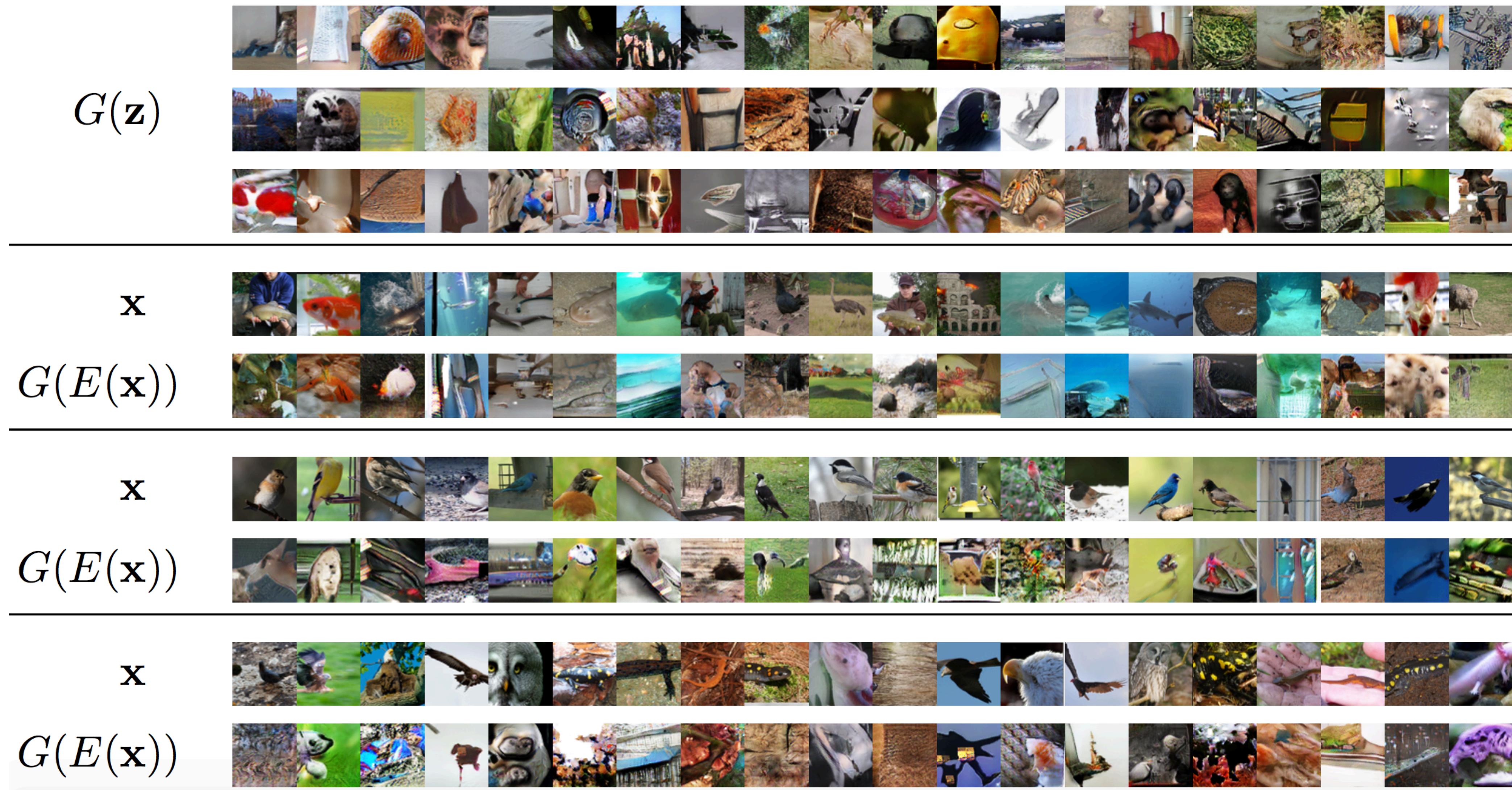


(c) Lighting

(d) Wide or Narrow



# BiGAN Results



# Wasserstein-GAN



Consider the distribution as if it were a block  
move the blocks to make the two distributions the same  
“Minimal effort”

$$\text{EMD}(p_{\text{data}}, p_g) = \inf_{\gamma \in \Pi(p_{\text{data}}, p_g)} \mathbb{E}_{(x,y) \sim \gamma} [| | x - y | |]$$

$$W(p_{\text{data}}, p_g) = \sup_{||f||_{L^1} \leq 1} \mathbb{E}_{x \sim p_{\text{data}}} [f(x)] - \mathbb{E}_{x \sim p_g} [f(x)]$$

# WGAN Results

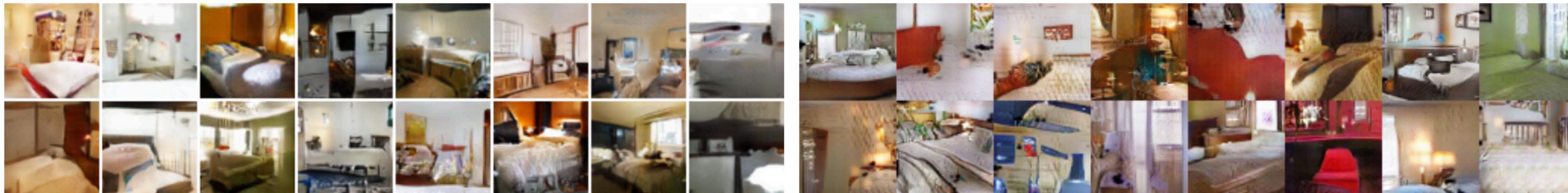


Figure 5: Algorithms trained with a DCGAN generator. Left: WGAN algorithm. Right: standard GAN formulation. Both algorithms produce high quality samples.



Figure 6: Algorithms trained with a generator without batch normalization and constant number of filters at every layer (as opposed to duplicating them every time as in [18]).

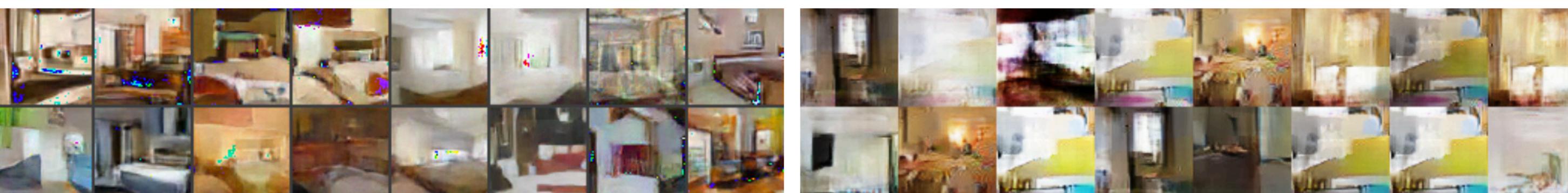


Figure 7: Algorithms trained with an MLP generator with 4 layers and 512 units with ReLU nonlinearities. The number of parameters is similar to that of a DCGAN, but it lacks a

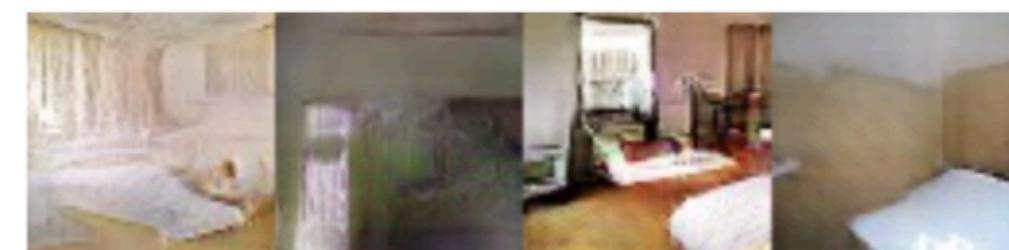
# WGAN-GP Results

**DCGAN**

Baseline ( $G$ : DCGAN,  $D$ : DCGAN)



**LSGAN**



**WGAN (clipping)**



**WGAN-GP (ours)**



$G$ : No BN and a constant number of filters,  $D$ : DCGAN



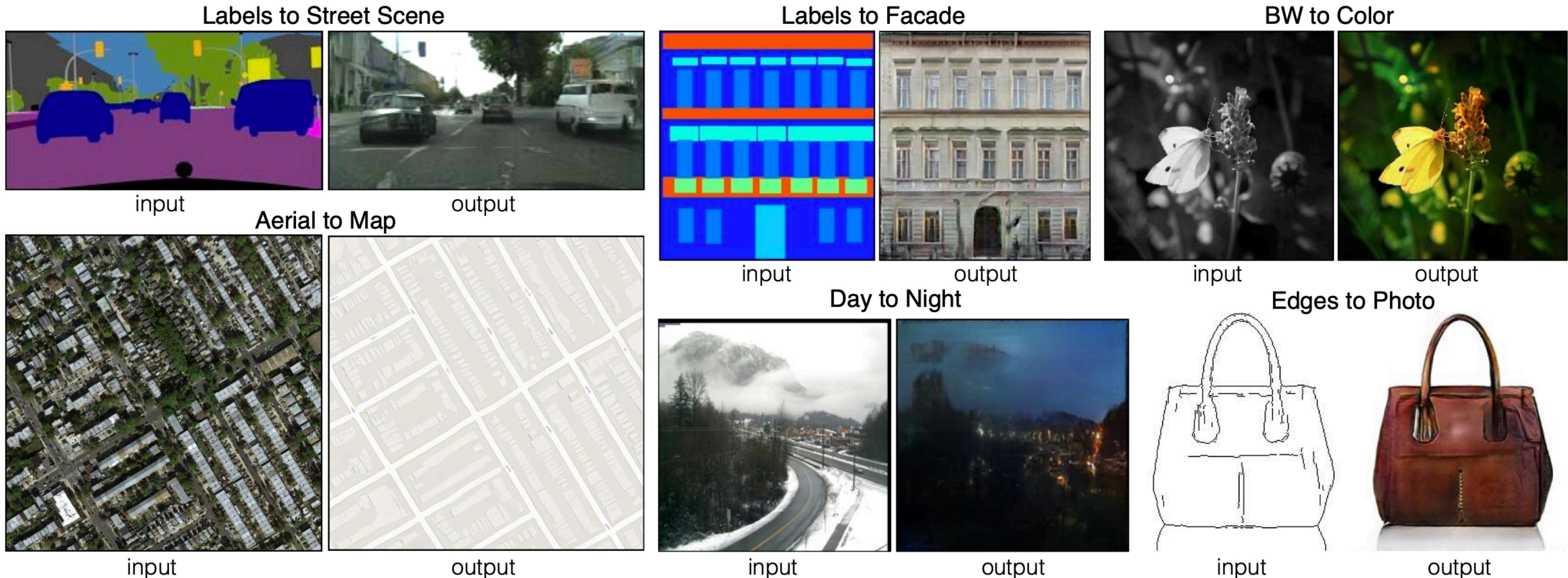
$G$ : 4-layer 512-dim ReLU MLP,  $D$ : DCGAN



No normalization in either  $G$  or  $D$



# Pix2Pix



# CycleGAN



# Style Transfer



Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge. "A neural algorithm of artistic style." *arXiv preprint arXiv:1508.06576* (2015).

# ProgressiveGAN results

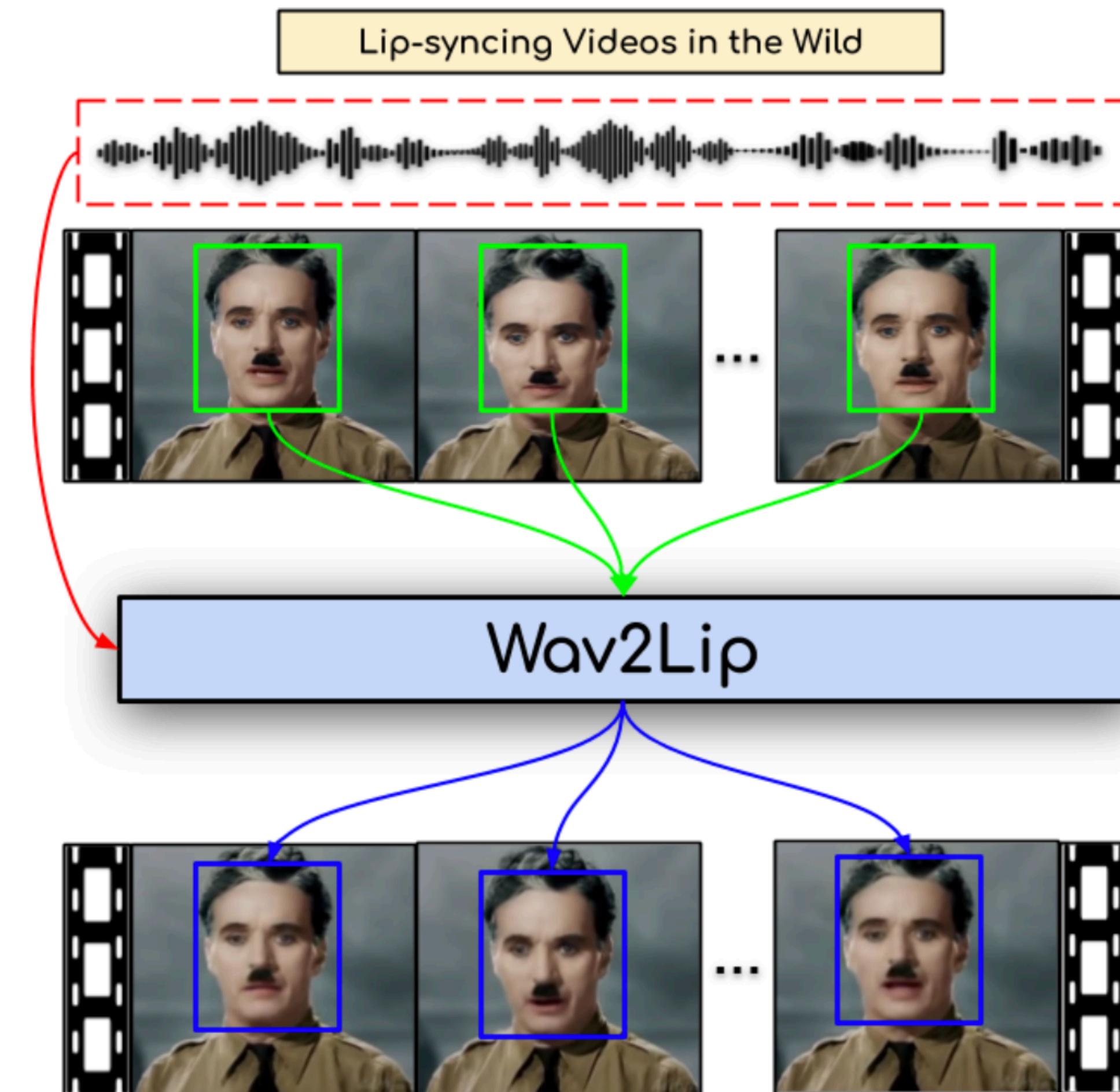
1024x1024 images generated using the CELEBA-HQ dataset.



# StyleGAN Coarse Styles



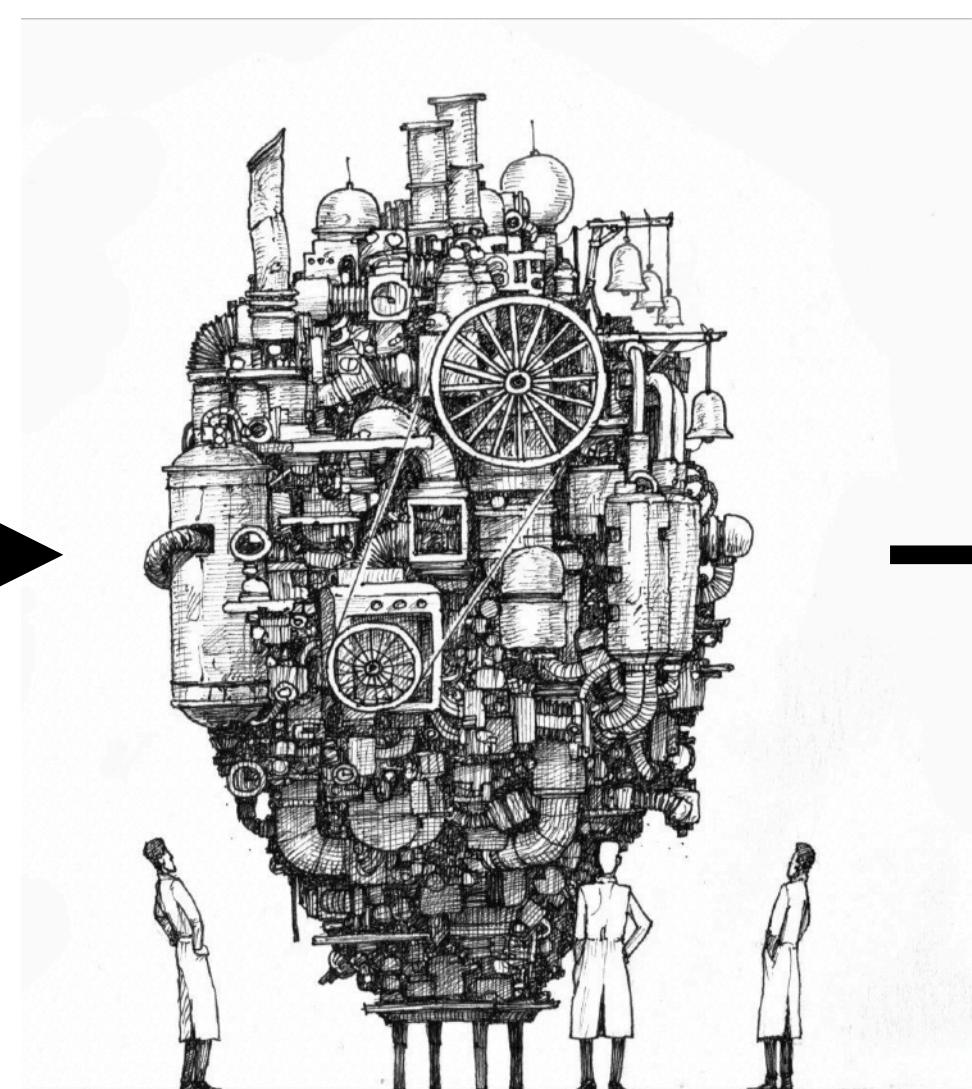
# Wav2Lip



- Demo : <https://youtu.be/0fXaDCZNOJc>

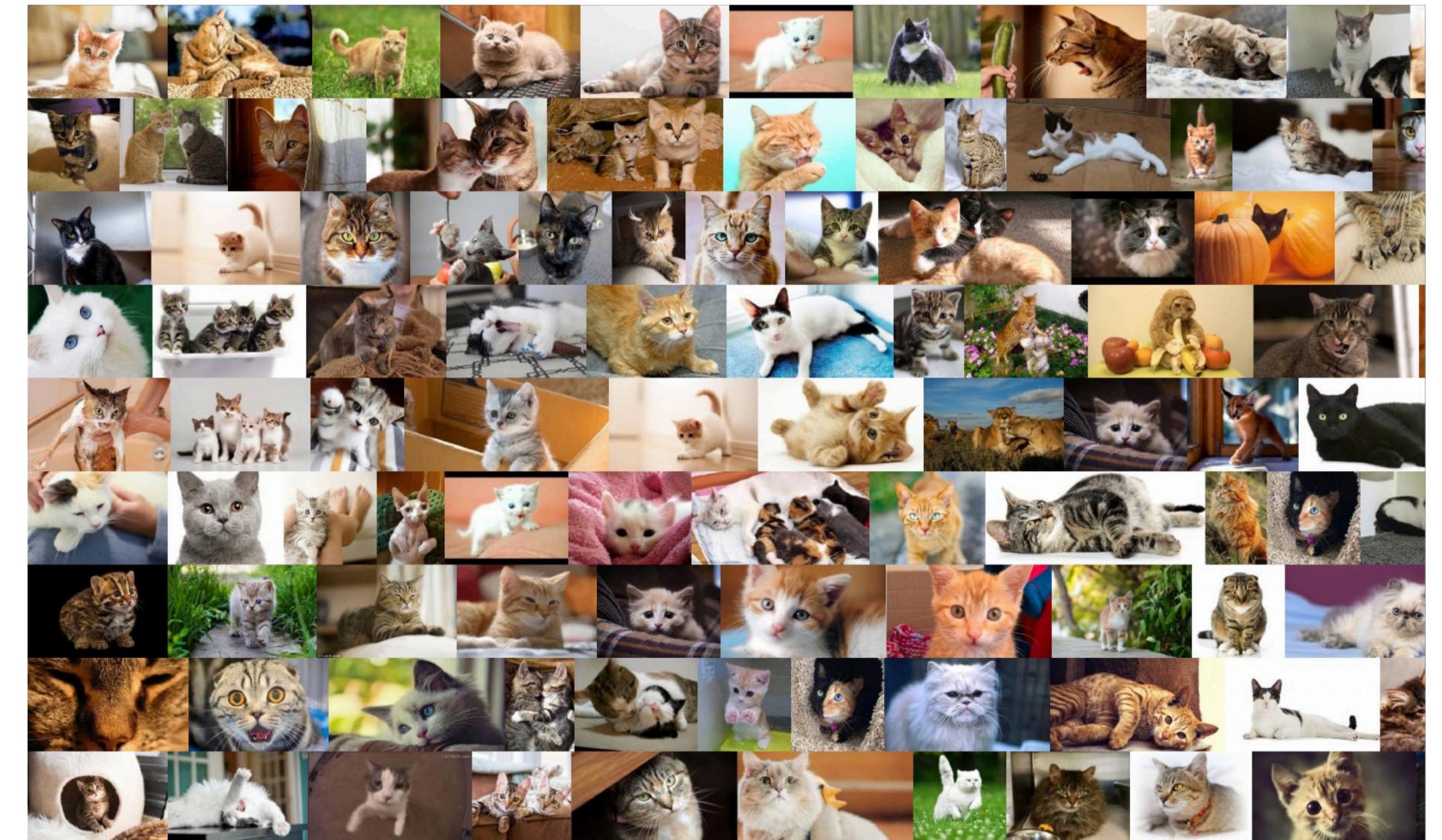
# Discriminative vs Generative Models

$$p(y|x)$$



“Cat”

$$p(x|y)$$



Discriminative Models

Generative Models

