Generative Models

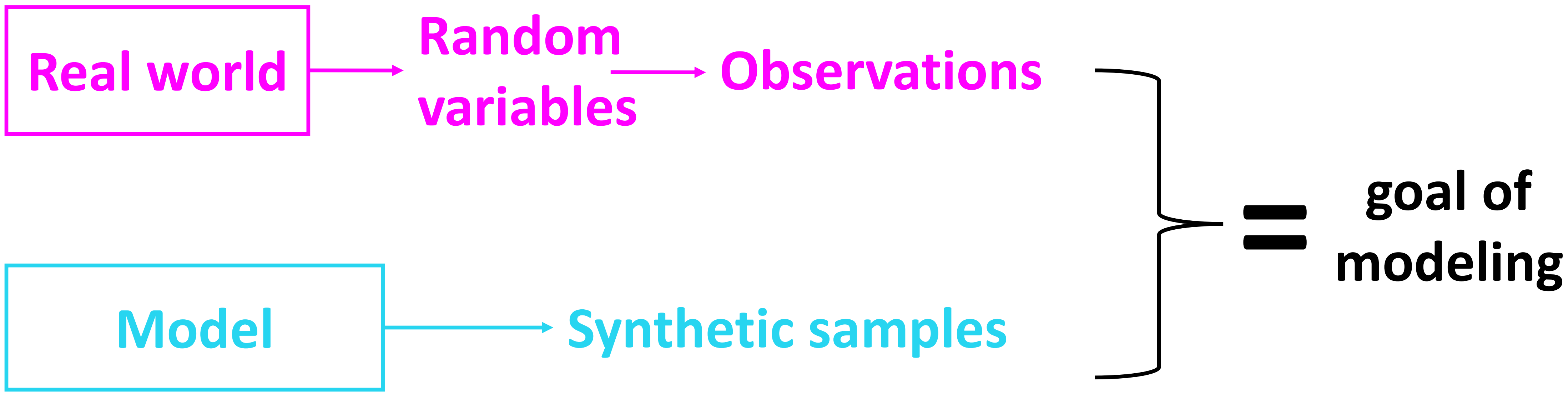
Dr. Kamal Mammadov

Part One: Generative Adversarial Networks

Overview

- GAN was first introduced by Ian Goodfellow et al in 2014
- Mainly be applied to generate images and videos
- Works well in continuous domain, but less well in discrete domains such as NLP

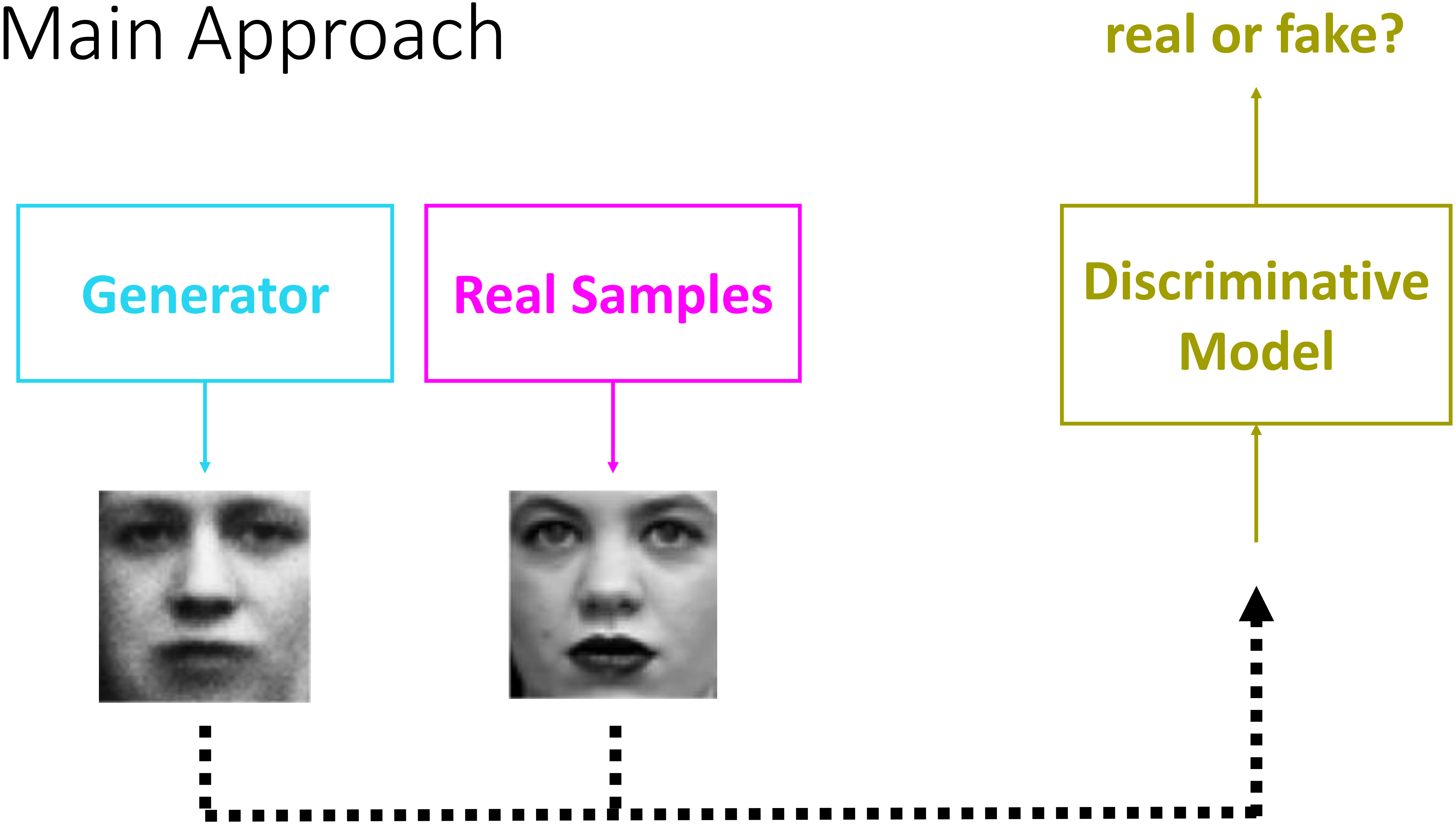
Goal of GAN



Density Estimation

- $Fit \, \Pr(Synthetic \; samples) = \Pr(Random \; Variables)$ using finite Observations

Main Approach



Main Approach

- Given a dataset $D = \{\mathbf{x}_i \big| i = 1, \cdots n\big\}$, where each x_i are independently sampled from an unknown distribution $p_d\big(\mathbf{x}\big)$.

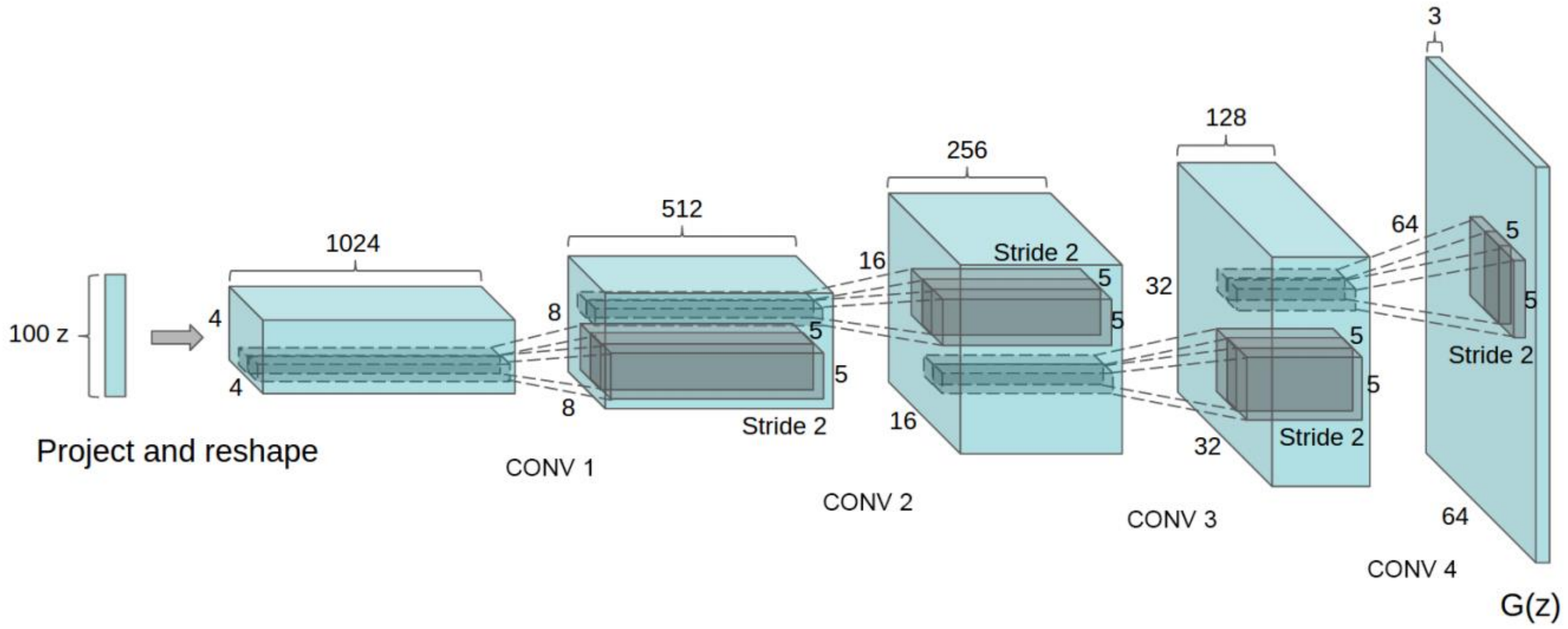
- Optimizing over discriminator
- $l_D(\Phi) = E_{z \sim p_D(x), z \sim p(z)}[-lnd(\mathbf{x}|\Phi) - ln(1 - d(g(\mathbf{z}|\Theta)|\Phi)].$
- Optimizing over generator
- $l_G(\Theta) = E_{z \sim p(z)}[ln(1 - d(g(\mathbf{z}|\Theta)|\Phi)].$

Theoretic Results and Practical difficulties

- Under certain assumptions, if the previous procedure achieves Nash equilibrium, the distribution formed by the generator is exactly the same as data distribution.

- The vanilla GAN, however, is hard to train, and has problem of diverging, model collapse, etc.

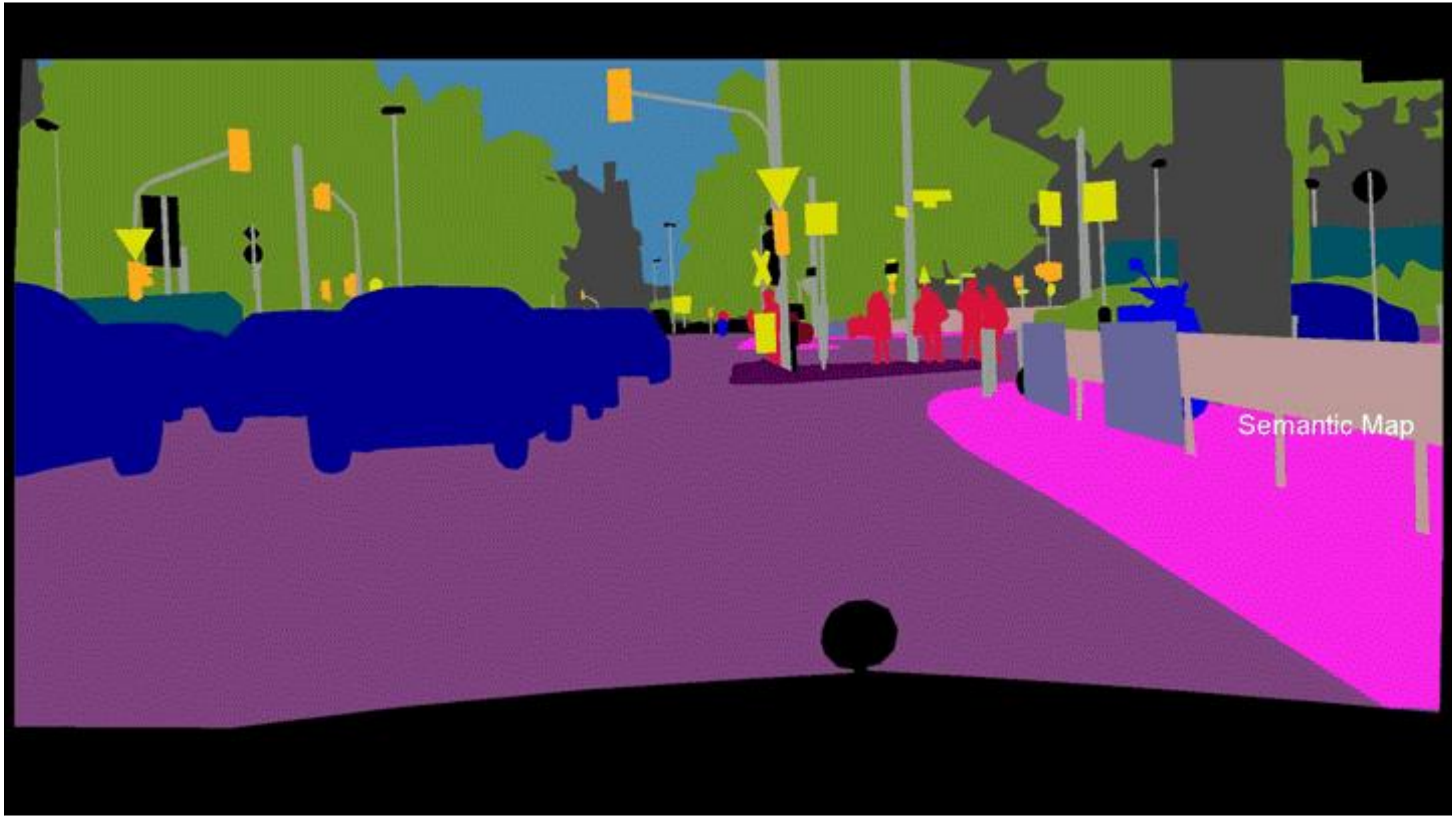
DCGAN


100 z
Project and reshape
1024
4
4
CONV 1
512
8
8
5
5
Stride 2
CONV 2
256
16
16
Stride 2
5
5
CONV 3
128
32
32
5
5
Stride 2
CONV 4
3
64
64
5
5
Stride 2
G(z)

DCGAN (Deep Convolutional GAN)

- Generates near real images with low resolutions
- Several tricks are used to stabilize the training of GAN
- https://github.com/carpedm20/DCGAN-tensorflow

High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs



High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs

Ting-Chun Wang[1] Ming-Yu Liu[1] Jun-Yan Zhu[2] Andrew Tao[1] Jan Kautz[1] Bryan Catanzaro[1]
[1]NVIDIA Corporation [2]UC Berkeley

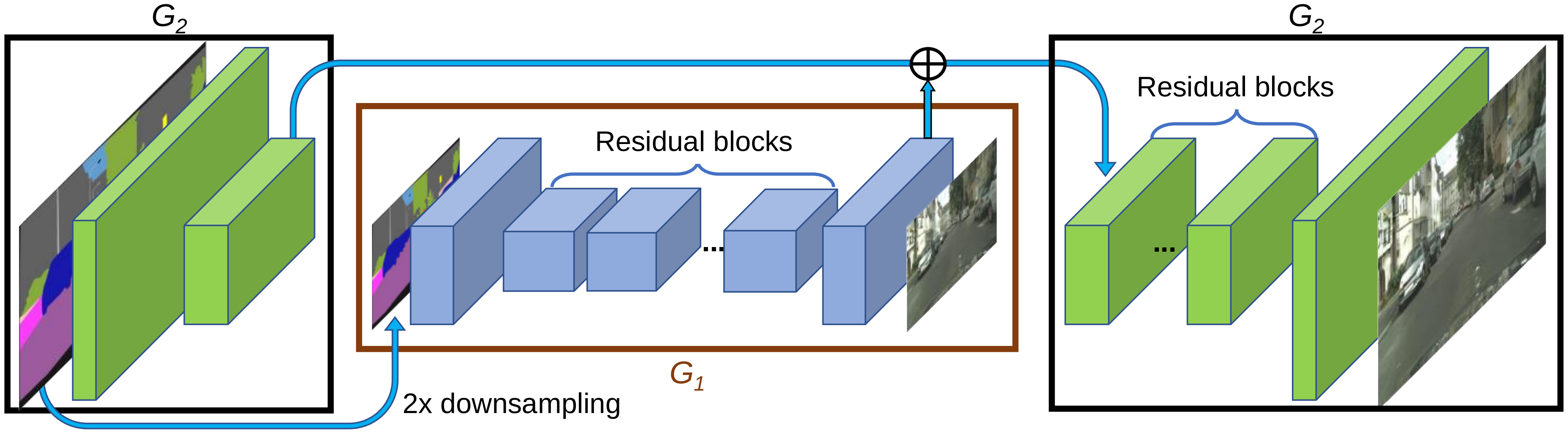


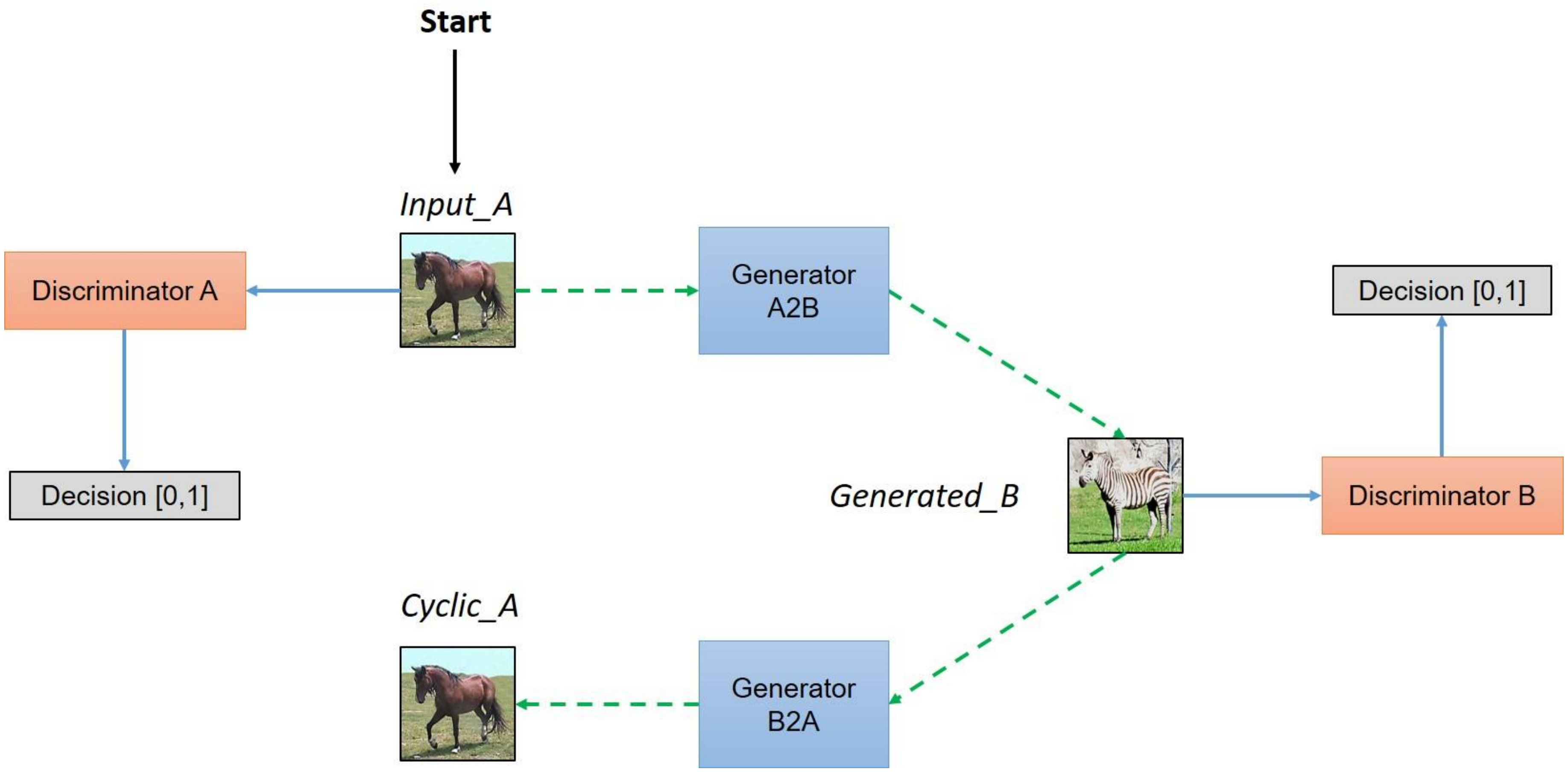
Figure 3: Network architecture of our generator. We first train a residual network G_1 on lower resolution images. Then, another residual network G_2 is appended to G_1 and the two networks are trained jointly on high resolution images. Specifically, the input to the residual blocks in G_2 is the element-wise sum of the feature map from G_2 and the last feature map from G_1.

Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks

Jun-Yan Zhu* Taesung Park* Phillip Isola Alexei A. Efros

UC Berkeley

In ICCV 2017

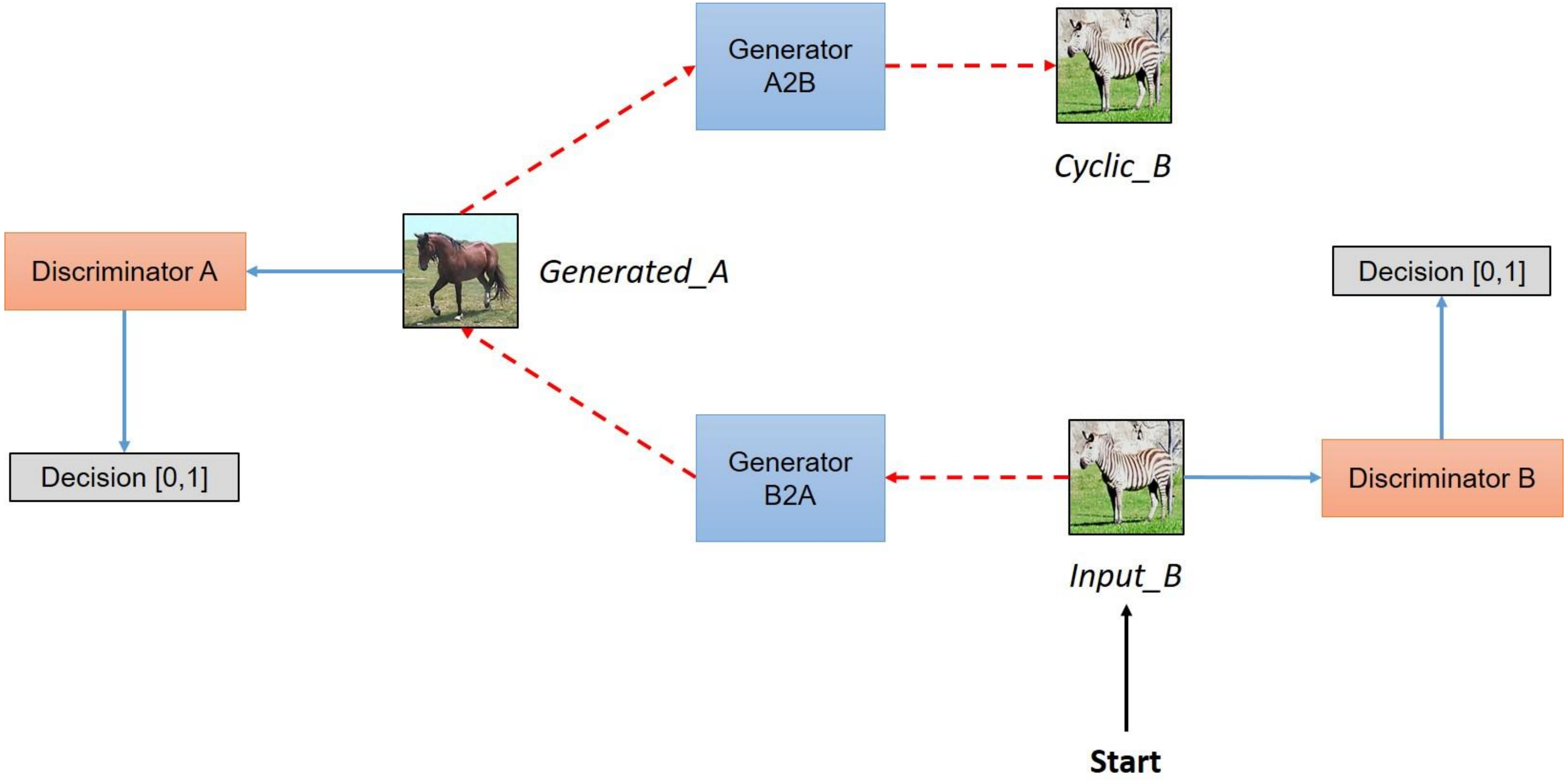


Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks

Jun-Yan Zhu* Taesung Park* Phillip Isola Alexei A. Efros

UC Berkeley

In ICCV 2017



Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks

Jun-Yan Zhu* **Taesung Park*** **Phillip Isola** **Alexei A. Efros**

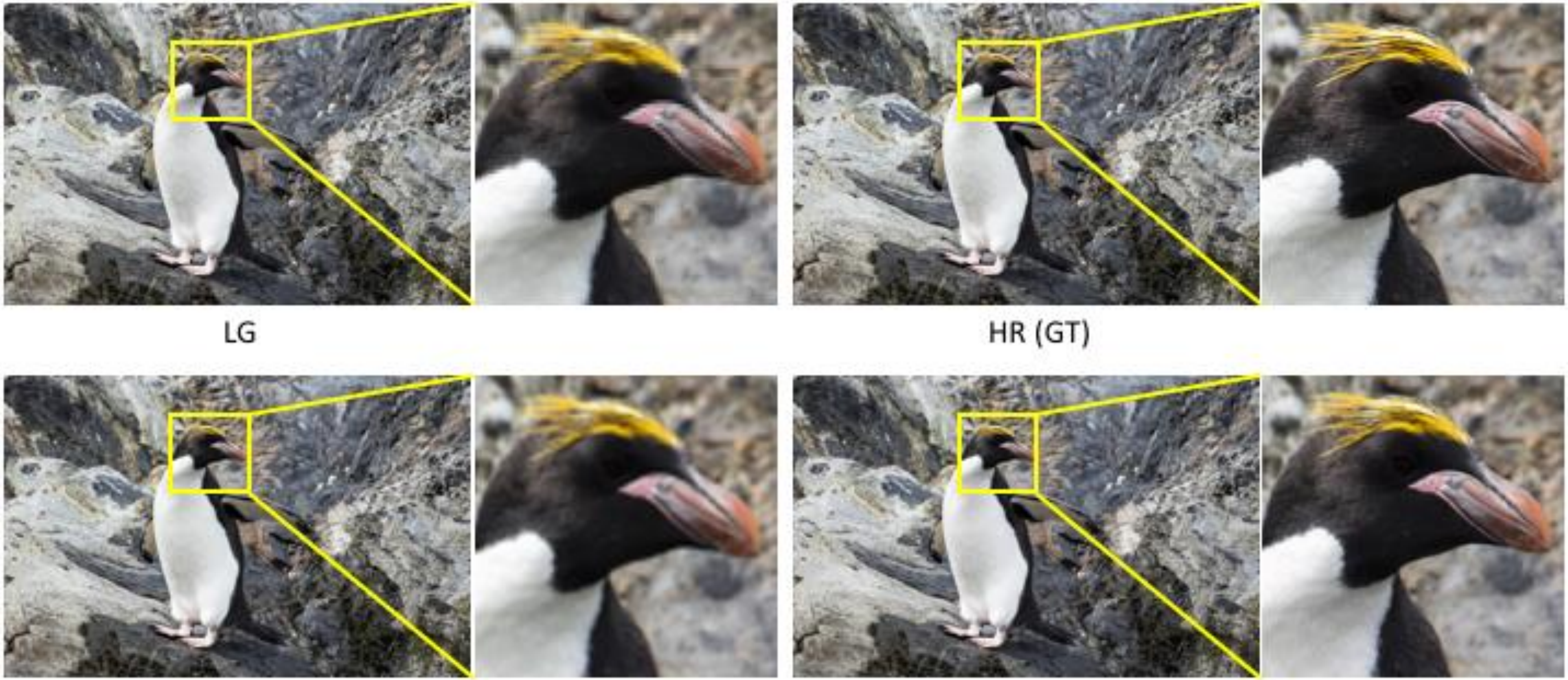
UC Berkeley

In ICCV 2017

GAN for Deblur (Deblur-GAN)

https://github.com/RaphaelMeudec/deblur-gan

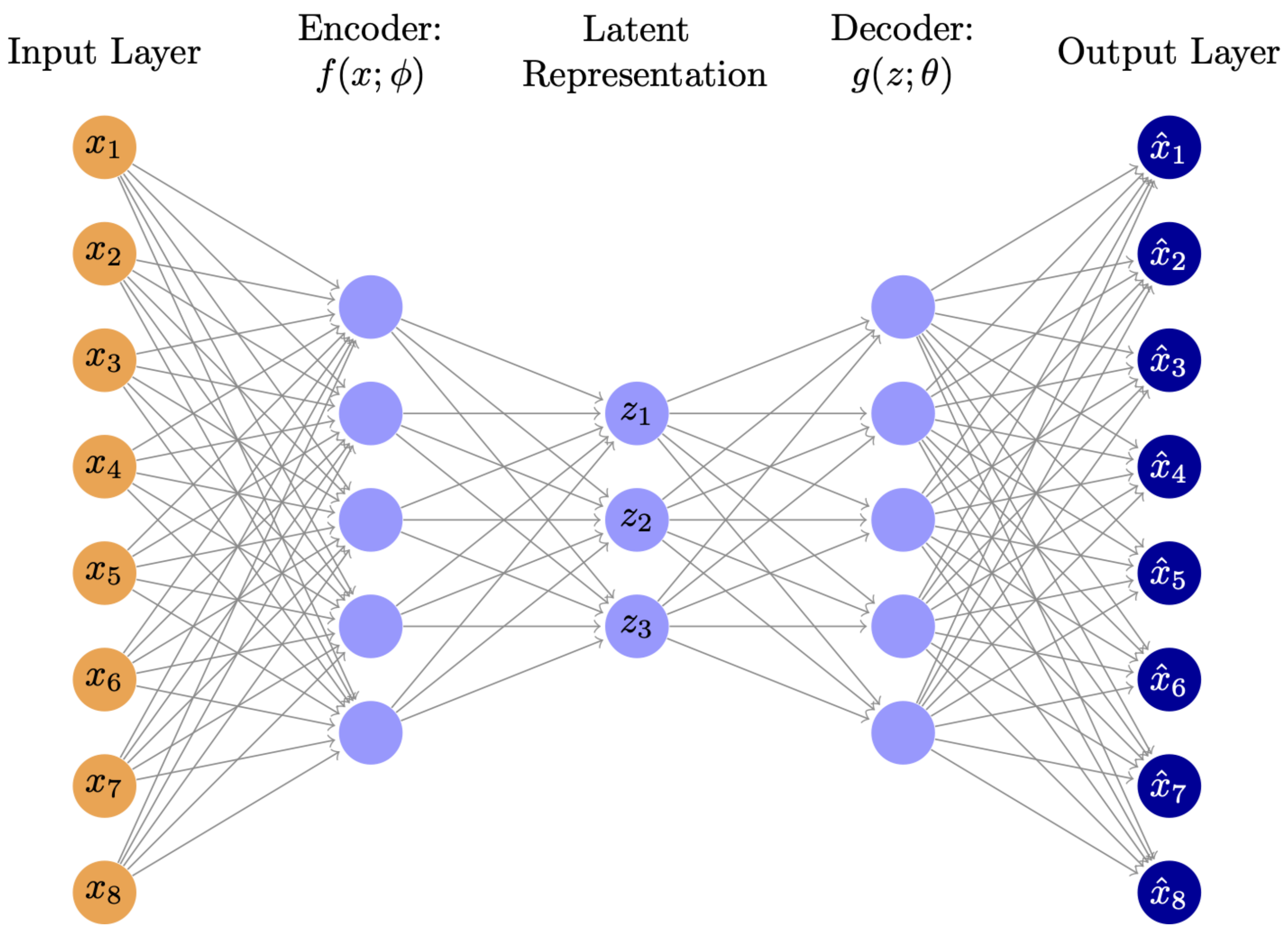
GAN for Super-Resolution (SRGAN)



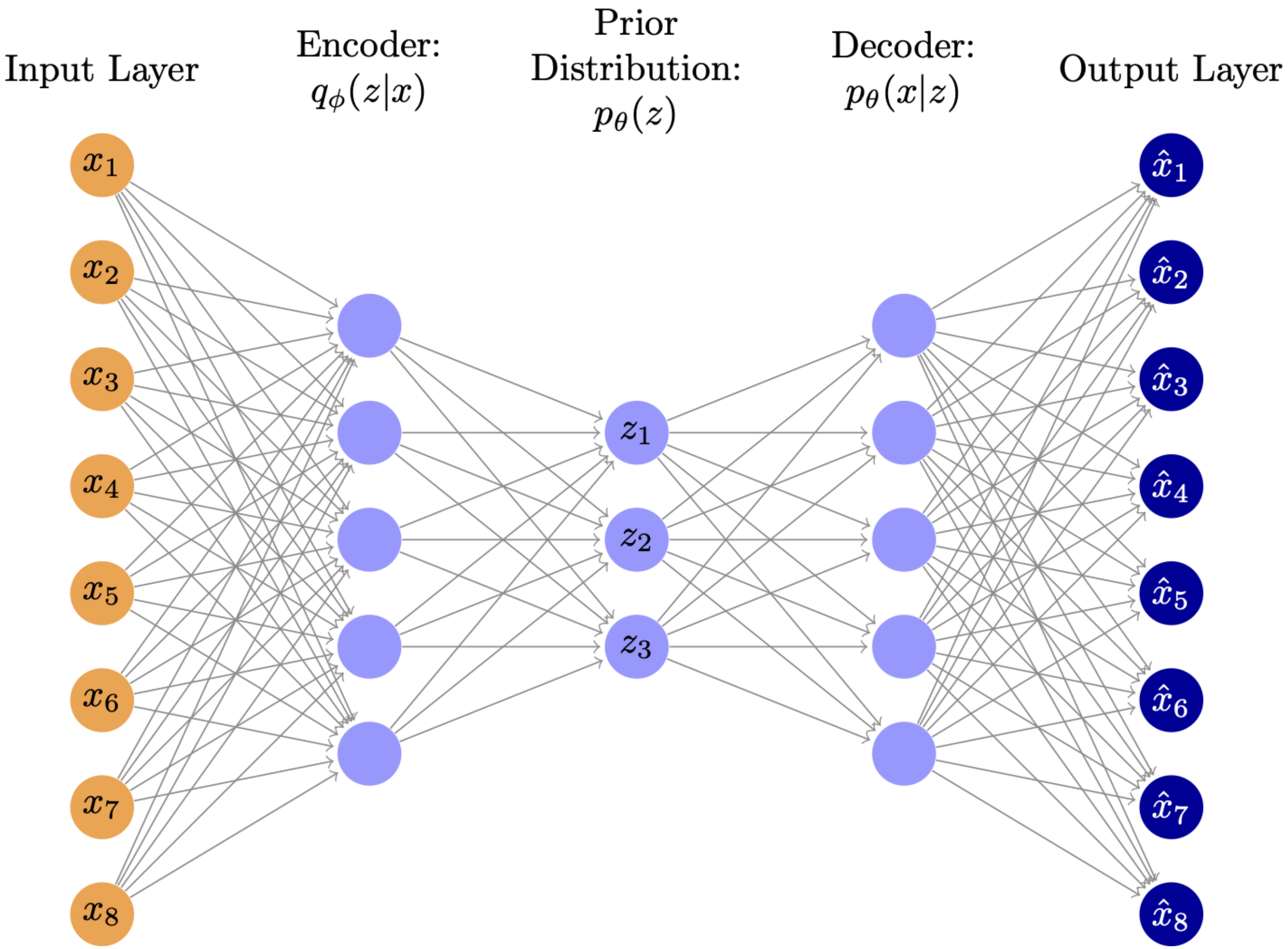
Part Two: Variational Auto Encoders

Overivew of Auto Encoder

- **Encoder:** $z = f(x; \phi)$
- **Decoder :** $\hat{x} = g(z; \theta)$
- **Loss Function :** $l = |x - g(f(x; \phi); \theta)|$,



Overview of Variational Auto Encoders



Basic Notations

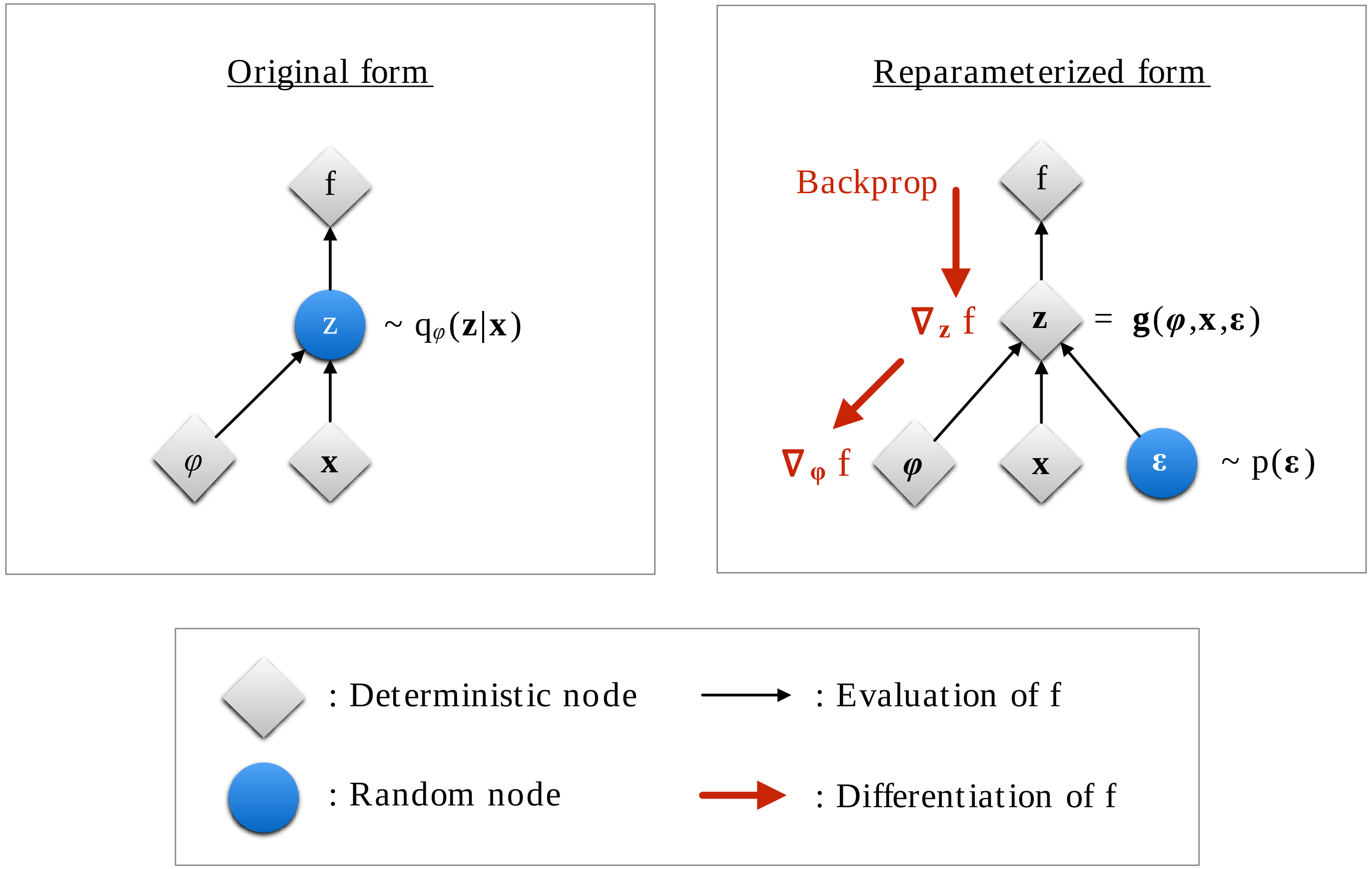
- Let $\{x|x \in X\}$ be a dataset with empirical distribution $q_D(x)$, the Variational Auto Encoder (VAE) is a generative model aimed at learning joint distribution
- $p_\theta(x, z) = p_\theta(x|z)p_\theta(z)$,
- where $p_\theta(z)$ is a prior distribution, and the encoder $q_\phi(\mathrm{z}|\mathrm{x})$ are used to approximate the true but intractable posterior $p_\theta(z|x)$.

The Evidence Lower Bound

$$\begin{aligned}
\log p_{\boldsymbol{\theta}}(\mathbf{x}) &= \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[\log p_{\boldsymbol{\theta}}(\mathbf{x})\right] \\
&= \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[\log\left[\frac{p_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{z})}{p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})}\right]\right] \\
&= \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[\log\left[\frac{p_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{z})}{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\frac{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}{p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})}\right]\right] \\
&= \underbrace{\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[\log\left[\frac{p_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{z})}{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\right]\right]}_{\substack{=\mathcal{L}_{\boldsymbol{\theta},\boldsymbol{\phi}}(\mathbf{x}) \\ \text{(ELBO)}}} + \underbrace{\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[\log\left[\frac{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}{p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})}\right]\right]}_{=D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})||p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}))}
\end{aligned}$$

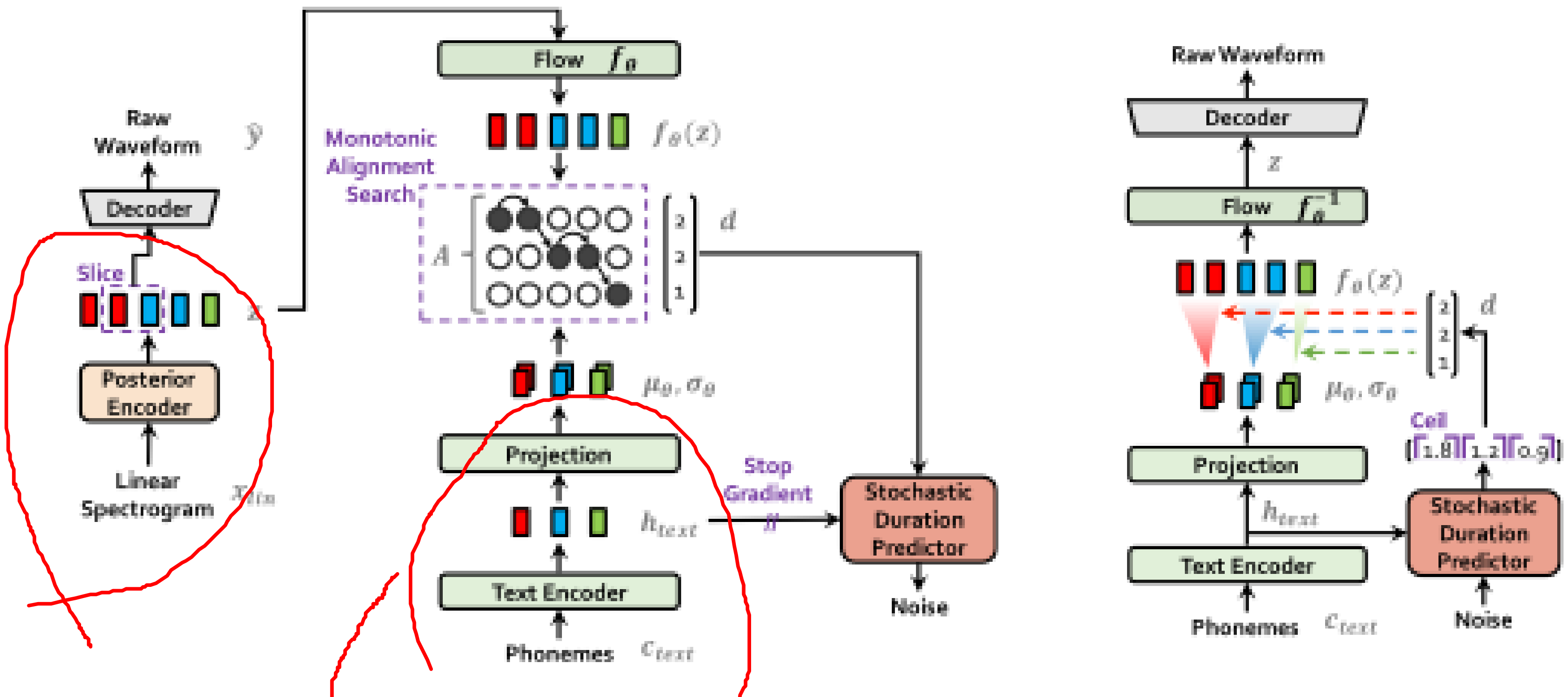
Reparameterisation Trick

- *Framework of the Reparameterization Trick. Source: Kingma, Diederik P., and Max Welling. "An introduction to variational autoencoders." Foundations and Trends® in Machine Learning 12.4 (2019): 307-392.*



Conditional Variational Autoencoder with Adversarial Learning for End-to-End Text-to-Speech

Jaehyeon Kim [1] **Jungil Kong** [1] **Juhee Son** [1 2]



(a) Training procedure

(b) Inference procedure

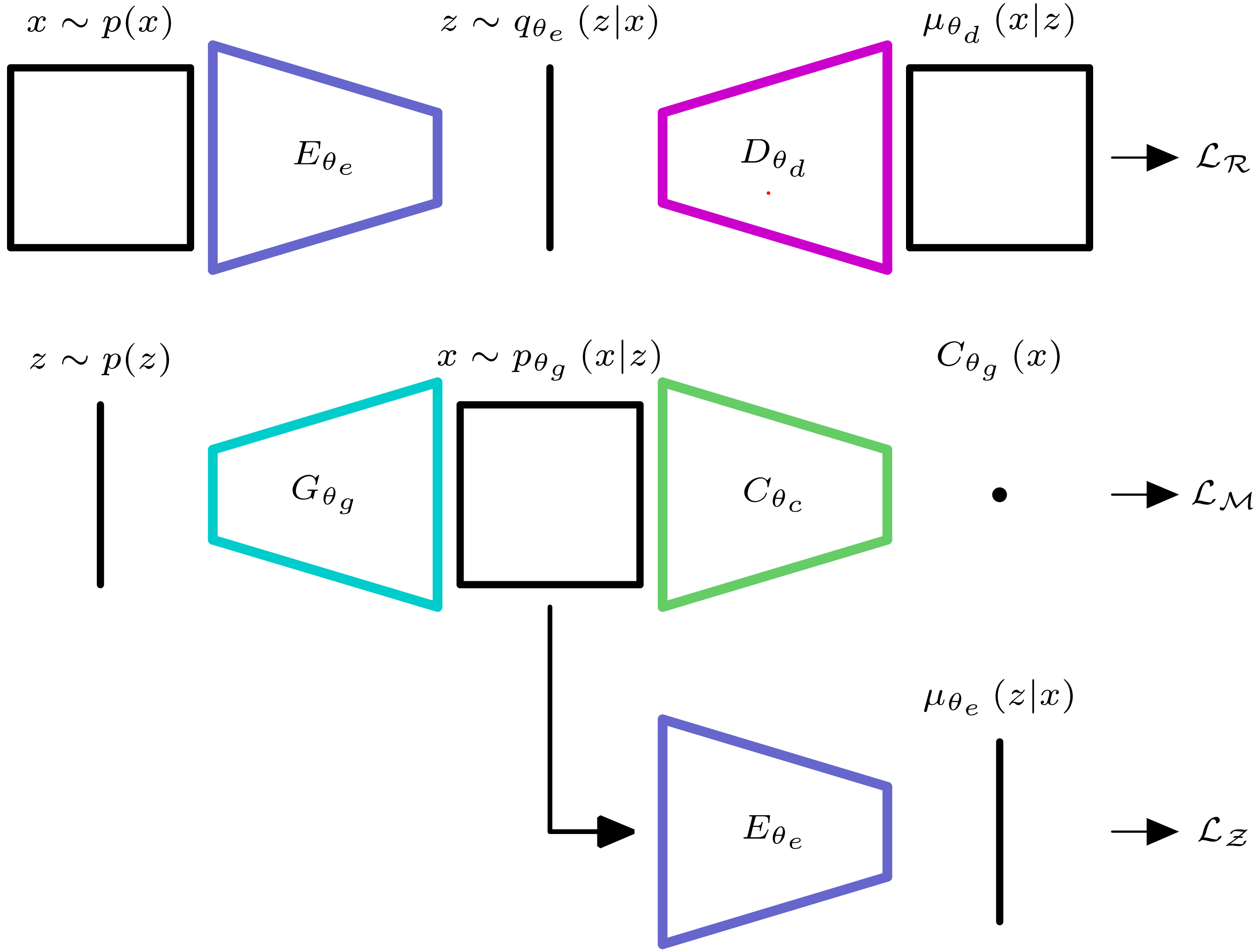
Figure 1. System diagram depicting (a) training procedure and (b) inference procedure. The proposed model can be viewed as a conditional VAE; a posterior encoder, decoder, and conditional prior (green blocks: a normalizing flow, linear projection layer, and text encoder) with a flow-based stochastic duration predictor.

AVAE: Adversarial Variational Auto Encoder*

Plumerault, Antoine[1,2], Le Borgne, Hervé[1], and Hudelot, Céline[2]

[1] *Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France , Email: {antoine.plumerault,herve.le-borgne}@cea.fr*

[2] *MICS, Centrale-Supelec, Gif-sur-Yvette, France , Email: celine.hudelot@centralesupelec.fr*



AVAE: Adversarial Variational Auto Encoder*

Plumerault, Antoine[1,2], Le Borgne, Hervé[1], and Hudelot, Céline[2]

[1] *Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France , Email: {antoine.plumerault,herve.le-borgne}@cea.fr*
[2] *MICS, Centrale-Supelec, Gif-sur-Yvette, France , Email: celine.hudelot@centralesupelec.fr*

		Bedroom	CelebA	CIFAR10	CIFAR100	SVHN
VAE	MSE	0.06 ± 0.00	0.03 ± 0.00	0.05 ± 0.00	0.05 ± 0.00	0.02 ± 0.00
	LPIPS	0.58 ± 0.00	0.18 ± 0.00	0.26 ± 0.00	0.25 ± 0.00	0.08 ± 0.00
	FID	229.75 ± 1.45	60.04 ± 0.47	136.75 ± 0.57	129.71 ± 1.01	68.16 ± 2.10
GAN	FID	110.59 ± 19.55	14.54 ± 0.41	32.01 ± 0.41	34.51 ± 0.59	23.83 ± 3.99
VAE/GAN	MSE	0.18 ± 0.01	0.07 ± 0.00	0.14 ± 0.02	0.15 ± 0.02	0.06 ± 0.02
	LPIPS	0.26 ± 0.01	0.09 ± 0.00	0.08 ± 0.01	0.08 ± 0.01	0.08 ± 0.02
	FID	60.02 ± 2.36	26.45 ± 4.66	39.04 ± 2.42	40.03 ± 0.71	17.02 ± 2.58
BiGAN	MSE	0.42 ± 0.05	0.18 ± 0.01	0.31 ± 0.02	0.33 ± 0.01	0.12 ± 0.01
	LPIPS	0.44 ± 0.02	0.16 ± 0.00	0.14 ± 0.00	0.16 ± 0.00	0.12 ± 0.01
	FID	91.72 ± 18.10	18.49 ± 5.06	34.61 ± 1.29	35.40 ± 1.23	27.77 ± 2.96
Ours with ξ with $\mathcal{L}^a_{\mathcal{Z}}$	MSE	0.12 ± 0.00	0.05 ± 0.00	0.09 ± 0.00	0.09 ± 0.00	0.04 ± 0.00
	LPIPS	0.36 ± 0.00	0.11 ± 0.00	0.10 ± 0.00	0.11 ± 0.00	0.10 ± 0.00
	FID	85.11 ± 2.87	16.99 ± 0.58	33.65 ± 0.28	39.81 ± 0.60	27.64 ± 2.41
Ours without ξ with $\mathcal{L}^a_{\mathcal{Z}}$	MSE	0.12 ± 0.00	0.05 ± 0.00	0.09 ± 0.00	0.09 ± 0.00	0.04 ± 0.00
	LPIPS	0.35 ± 0.00	0.11 ± 0.00	0.10 ± 0.00	0.11 ± 0.00	0.09 ± 0.00
	FID	84.29 ± 5.28	16.23 ± 0.50	33.49 ± 0.50	38.69 ± 0.62	28.47 ± 8.24
Ours without ξ with $\mathcal{L}^b_{\mathcal{Z}}$	MSE	0.12 ± 0.00	0.05 ± 0.00	0.09 ± 0.00	0.09 ± 0.00	0.04 ± 0.00
	LPIPS	0.35 ± 0.00	0.11 ± 0.00	0.10 ± 0.00	0.11 ± 0.00	0.08 ± 0.00
	FID	80.99 ± 1.82	15.01 ± 0.82	33.67 ± 0.61	38.35 ± 0.57	21.11 ± 0.42

Table 1: Reconstruction errors (MSE and LPIPS [31]) and FID [10] of generated images for different models. Lower values are better for all metrics. Reported results are the average and standard deviation over five runs.