Institut
des algorithmes
d'apprentissage
de Montréal

MILA

Deep Generative Models

Aaron Courville

MILA, Université de Montréal

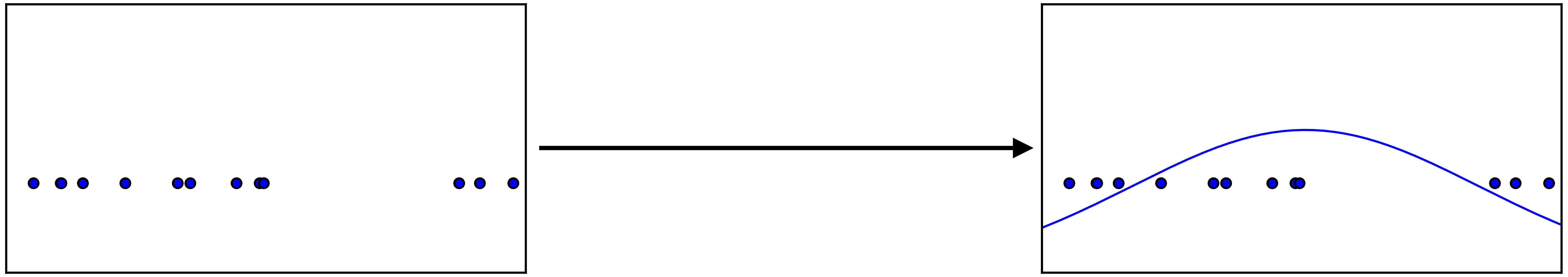
6.S191: Introduction to Deep Learning

MIT, Jan 30th, 2018

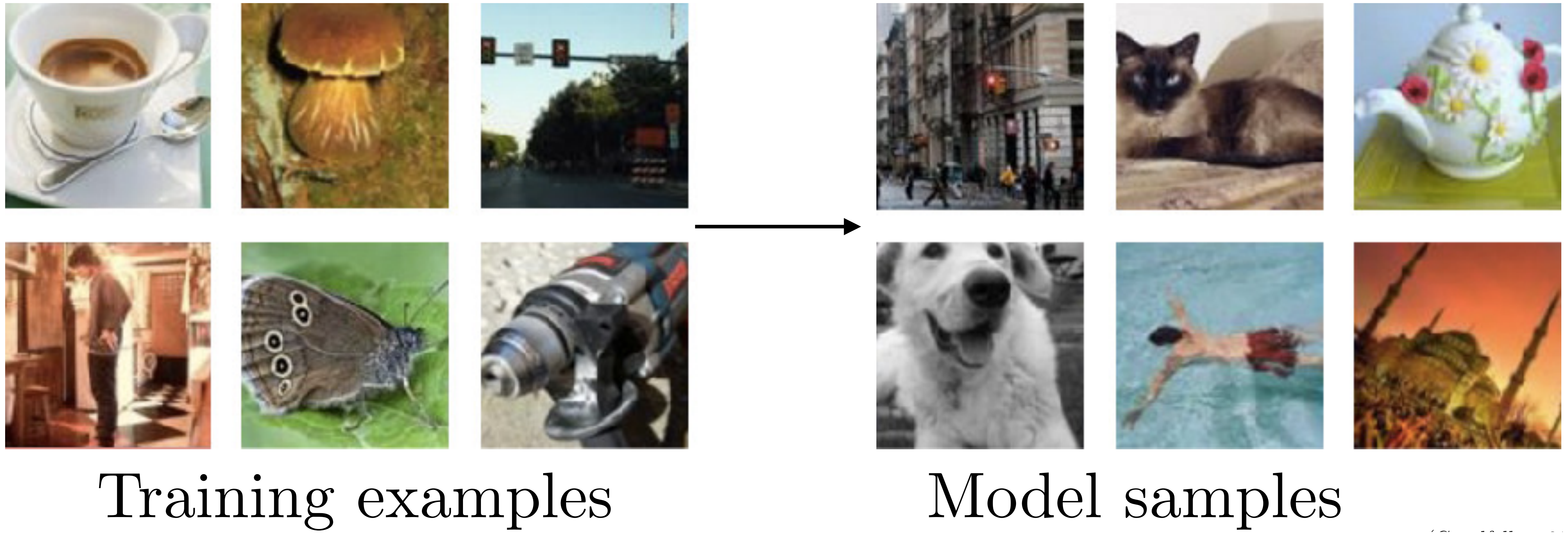
Generative modeling

- Generative models take training samples from some data distribution and learn a model that represents that distribution.

- Density estimation:



- Sample generation:



images taken from Goodfellow (2017)

Why generative models?

- Many tasks require structured output
  - Eg. Machine translation

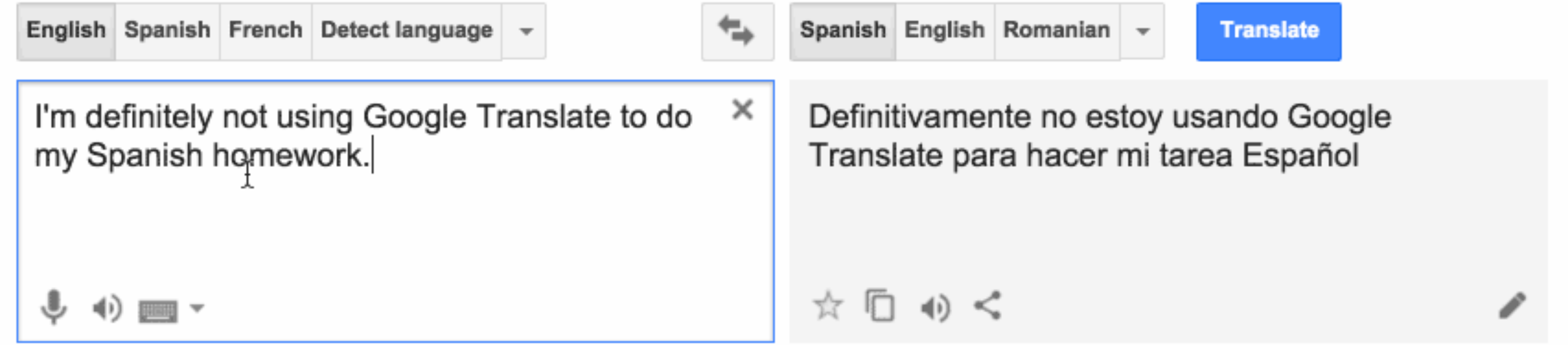
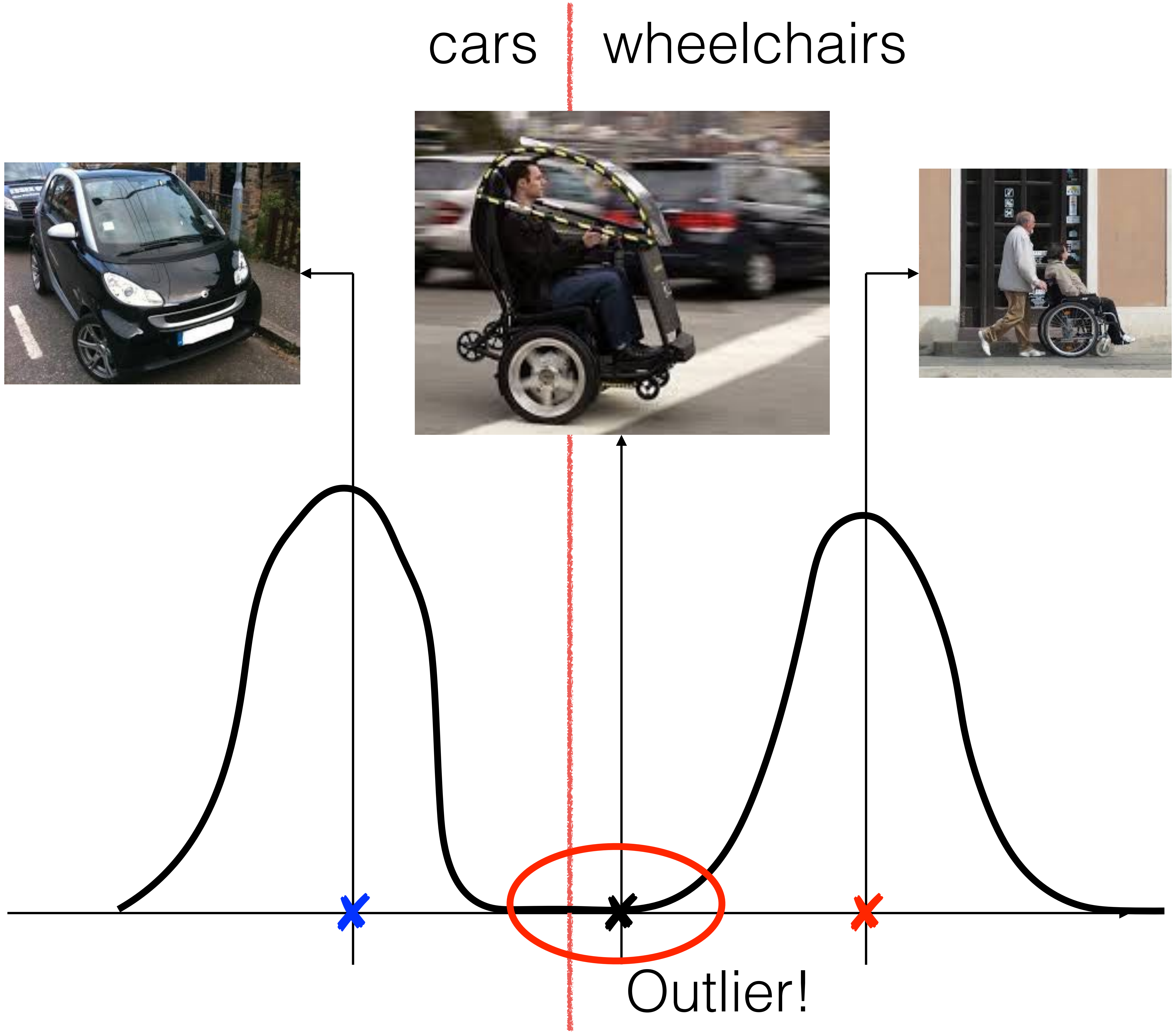


image credit: Adam Geitgey blog (2016) *Machine Learning is Fun Part 5: Language Translation with Deep Learning and the Magic of Sequences*

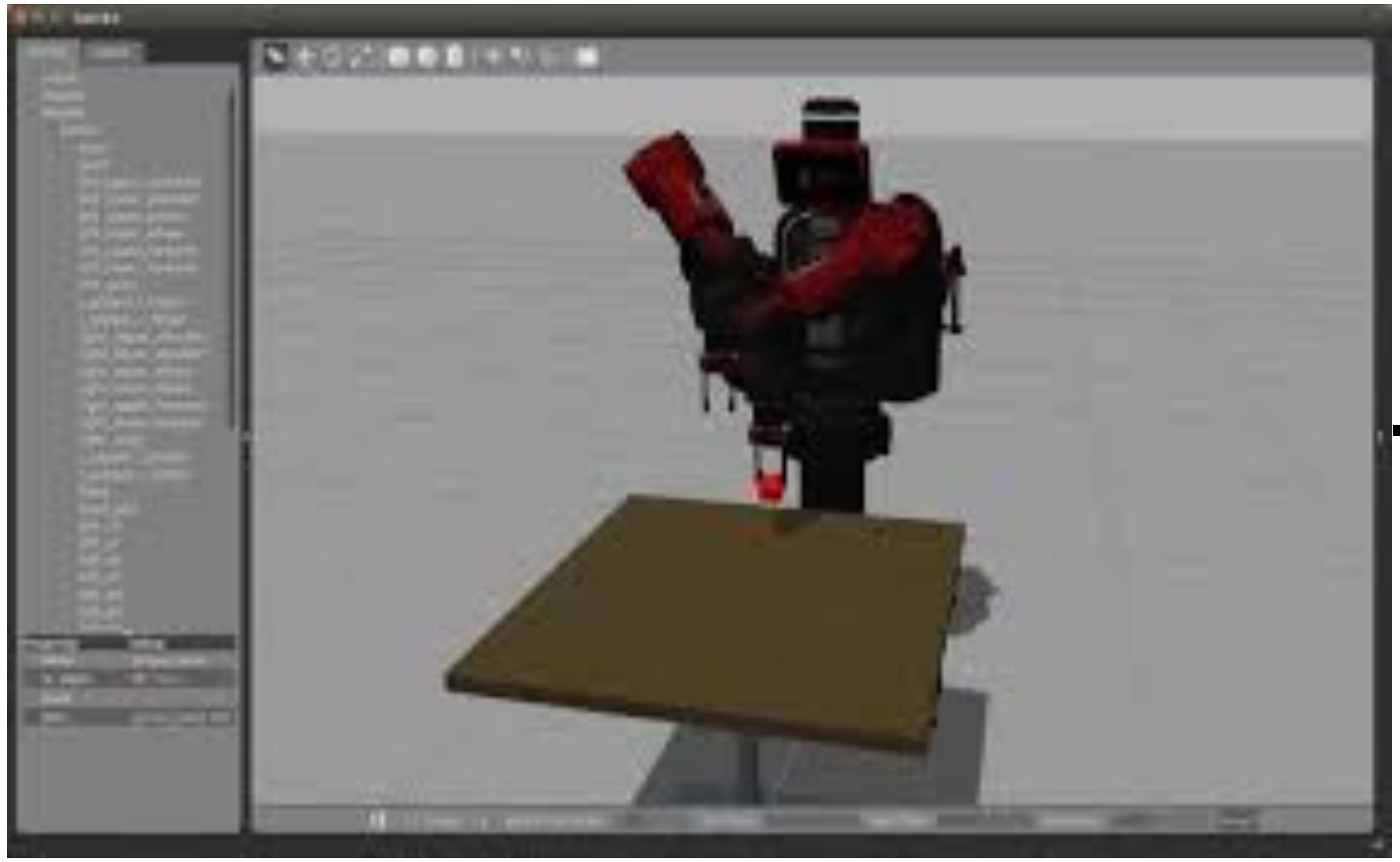
Why Generative Models? Outlier detection

- Large-scale deployment of CNN-based perception systems is becoming a reality.
- How do we detect when we encounter something new or rare (i.e. not appearing in the training data)?
- Goal: detect these outliers (anomalies) to avoid dangerous misclassification.
- Strategy: Leverage generative models of the training distribution to detect outliers.



Why Generative Models? Generation for Simulation

- Supports Reinforcement Learning for Robotics: Make simulations sufficiently realistic that learned policies can readily transfer to real-world application



Generative model →

Photo from IEEE Spectrum

Deep Generative Models: Outline

Autoregressive models

- Deep NADE, PixelRNN, PixelCNN, WaveNet, Video Pixel Network, etc.

Latent variable models

- Variational Auto encoders
- Generative Adversarial Networks

our focus today

Latent Variable Models

- The Variational Autoencoder model:
  - Kingma and Welling, *Auto-Encoding Variational Bayes*, *International Conference on Learning Representations (ICLR)* 2014.
  - Rezende, Mohamed and Wierstra, *Stochastic back-propagation and variational inference in deep latent Gaussian models*. ICML 2014.

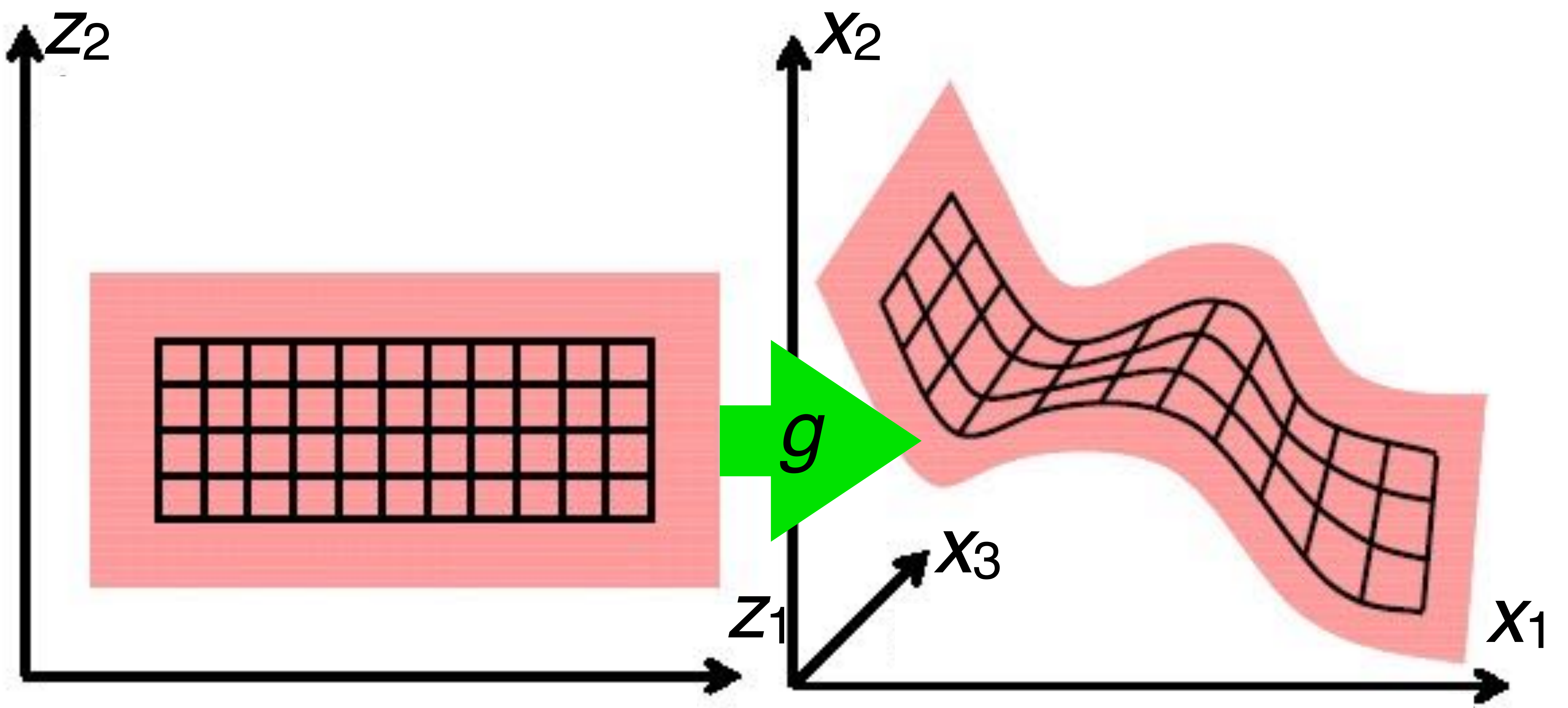
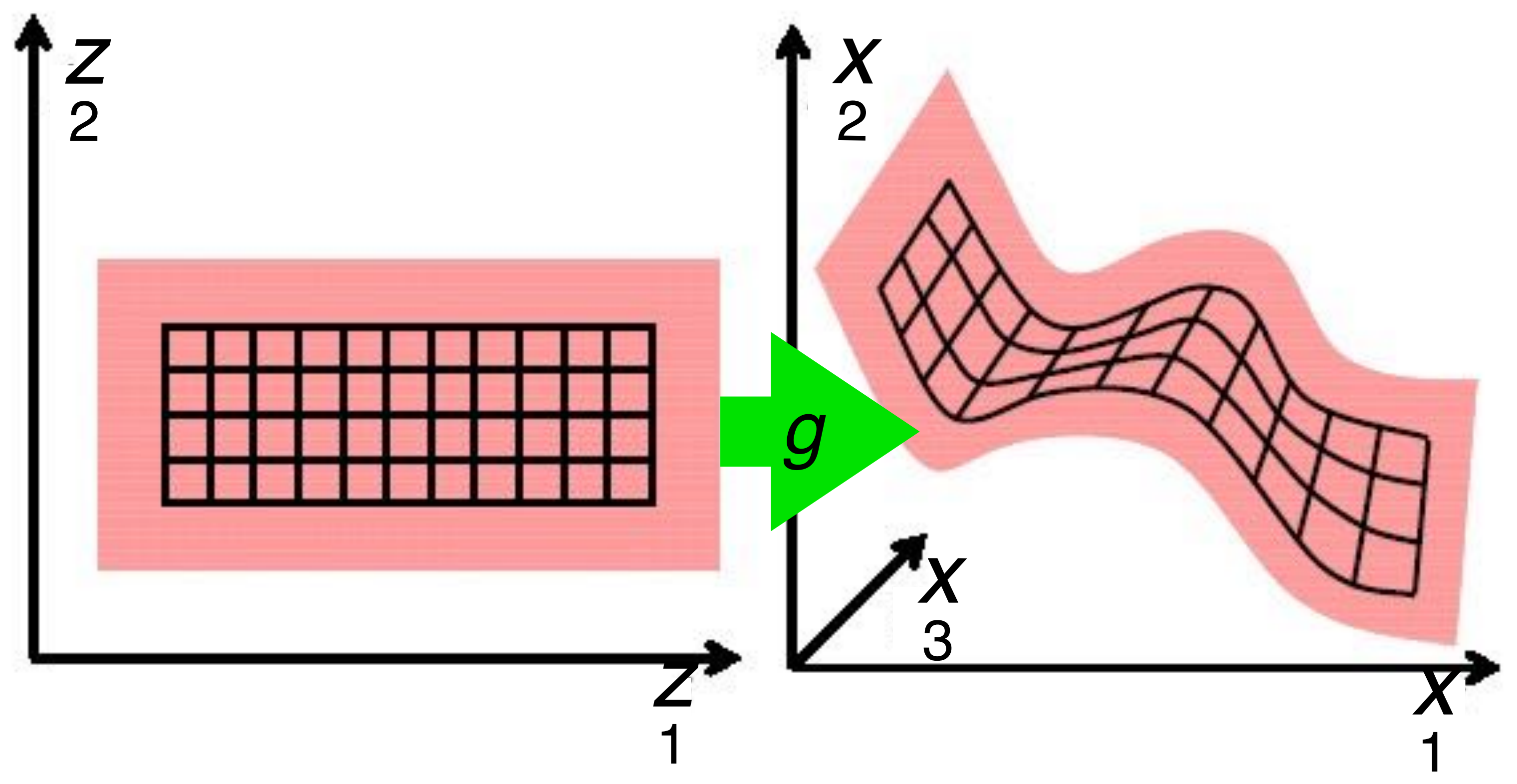
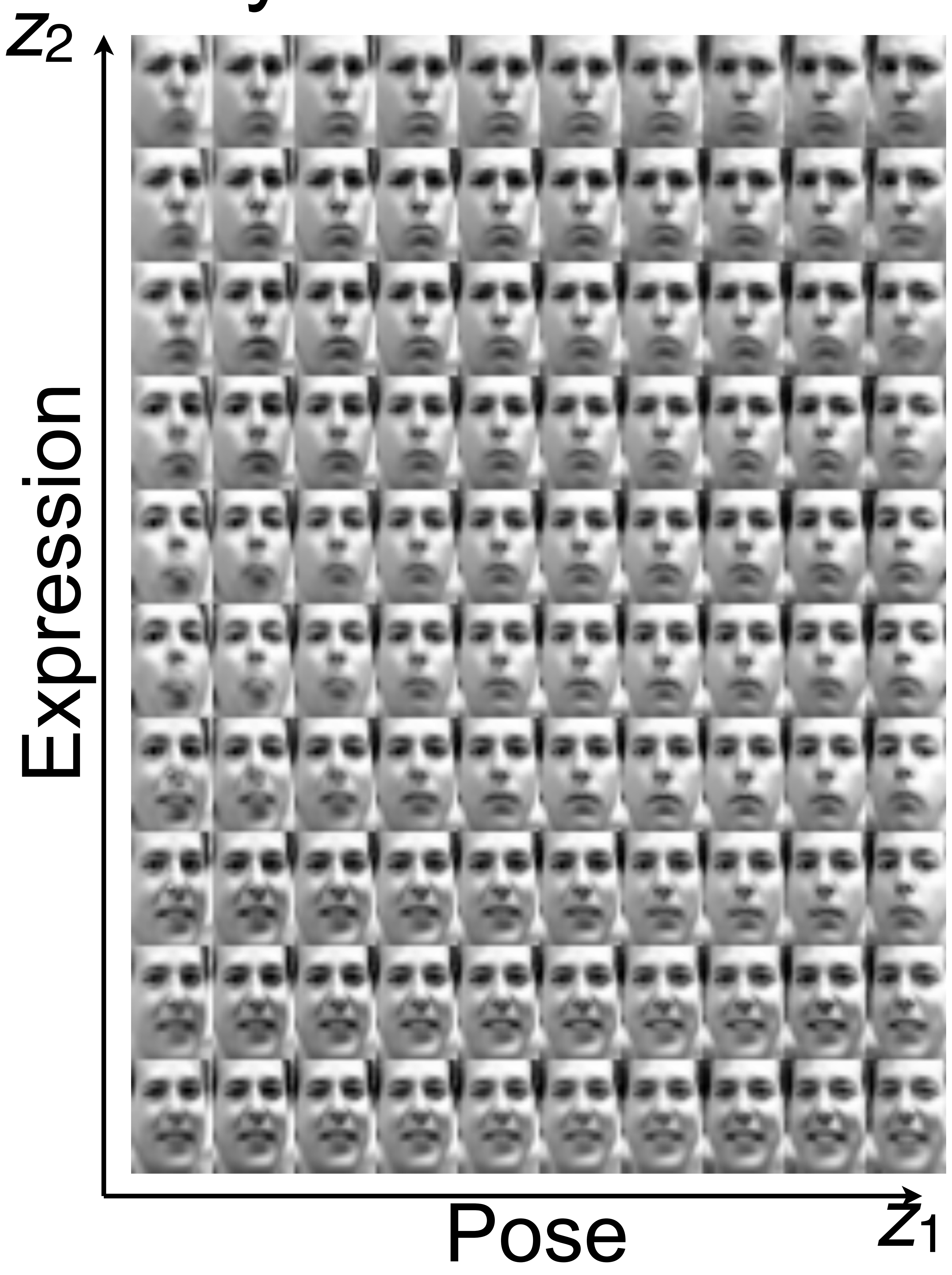


Image from: Ward, A. D., Hamarneh, G.: **3D Surface Parameterization Using Manifold Learning for Medial Shape Representation**, *Conference on Image Processing, Proc. of SPIE Medical Imaging*, 2007

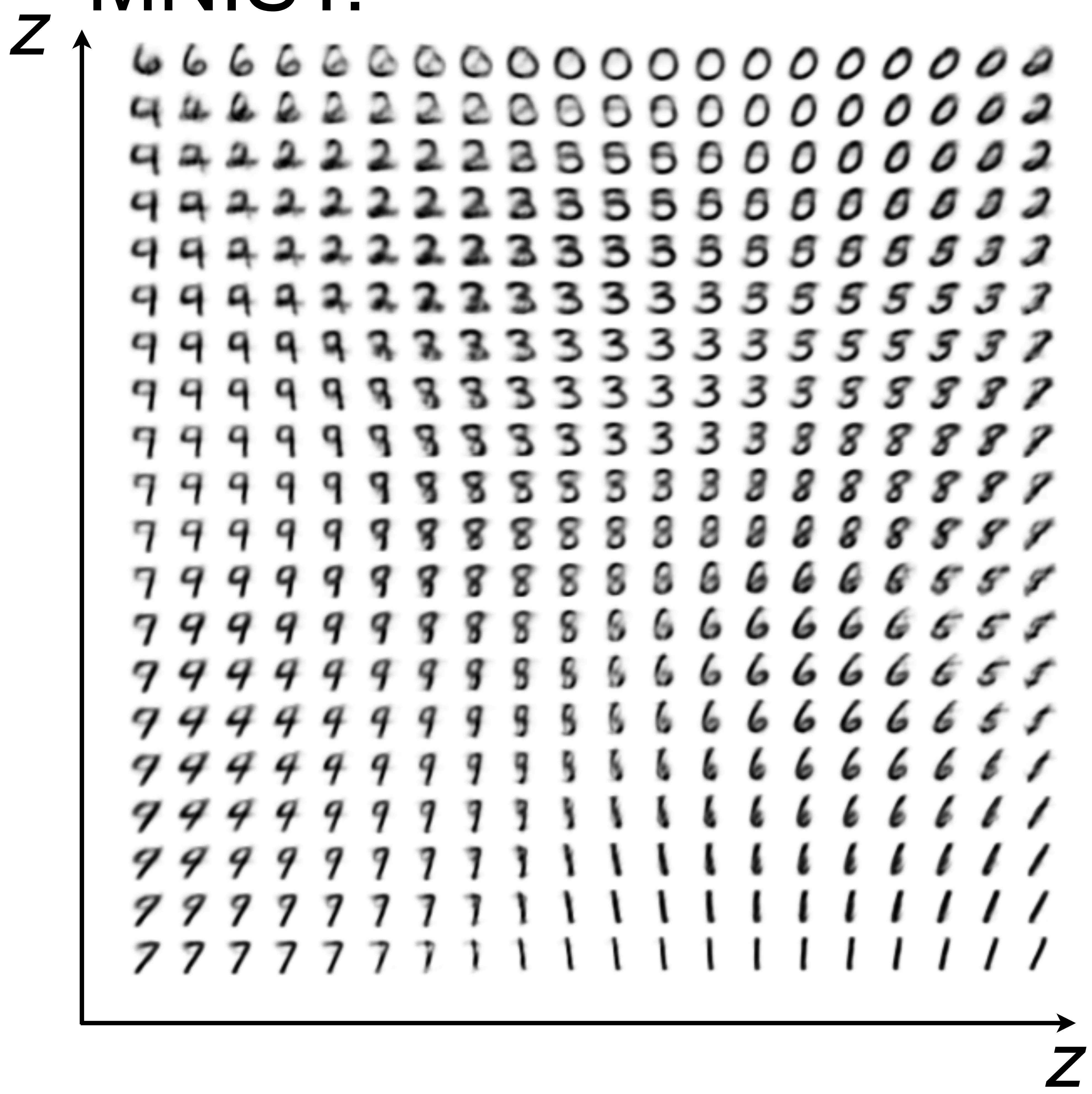
Latent Variable Models



Frey Faces:



MNIST:



Latent Variable Models

- latent variable model: learn a mapping from some latent variable z to a complicated distribution on x.

$$p(x) = \int p(x,z)\ dz \qquad \text{where} \quad p(\boldsymbol{x},\boldsymbol{z}) = p(\boldsymbol{x} \mid \boldsymbol{z})p(\boldsymbol{z})$$

Prior $$p(\boldsymbol{z}) = \text{something simple} \qquad p(\boldsymbol{x} \mid \boldsymbol{z}) = g(\boldsymbol{z})$$

- Can we learn to decouple the true explanatory factors underlying the data distribution? E.g. separate identity and expression in face images

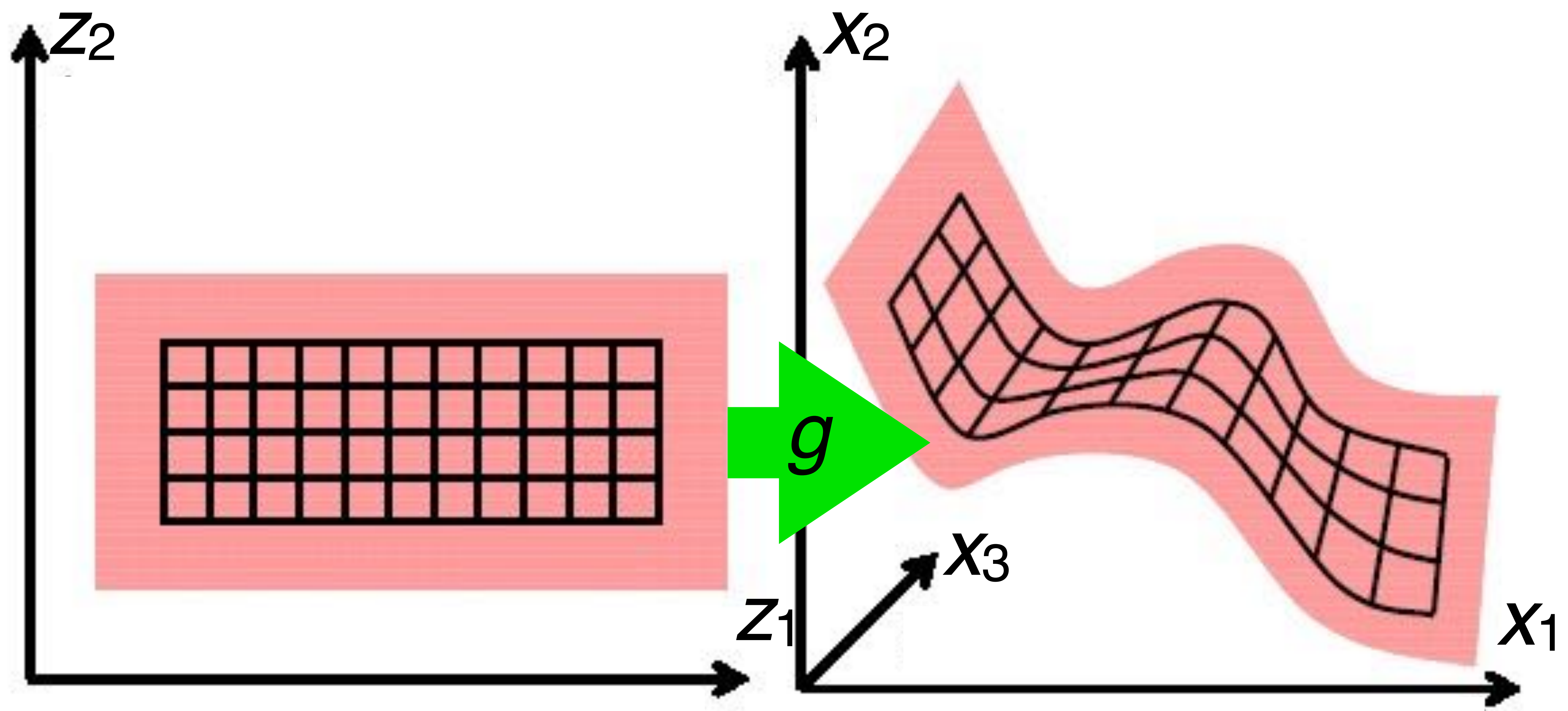


Image from: Ward, A. D., Hamarneh, G.: **3D Surface Parameterization Using Manifold Learning for Medial Shape Representation**, *Conference on Image Processing, Proc. of SPIE Medical Imaging*, 2007

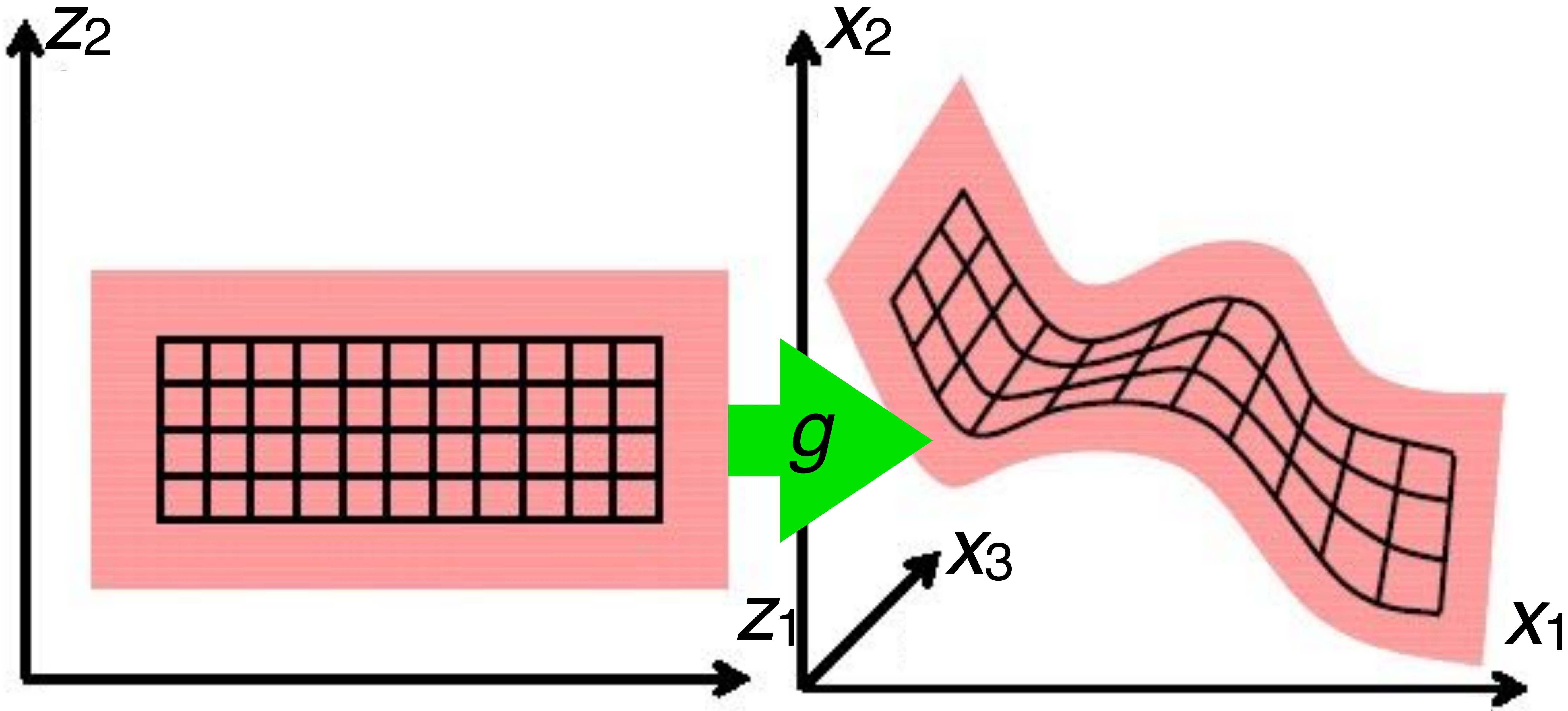
Latent Variable Models

- latent variable model: learn a mapping from some latent variable z to a complicated distribution on x.

$$p(x) = \int p(x,z)\; dz \qquad \text{where} \quad p(\boldsymbol{x},\boldsymbol{z}) = p(\boldsymbol{x} \mid \boldsymbol{z})p(\boldsymbol{z})$$

Prior $p(\boldsymbol{z}) = \text{something simple} \qquad p(\boldsymbol{x} \mid \boldsymbol{z}) = g(\boldsymbol{z})$

- Can we learn to decouple the true explanatory factors underlying the data distribution? E.g. separate identity and expression in face images



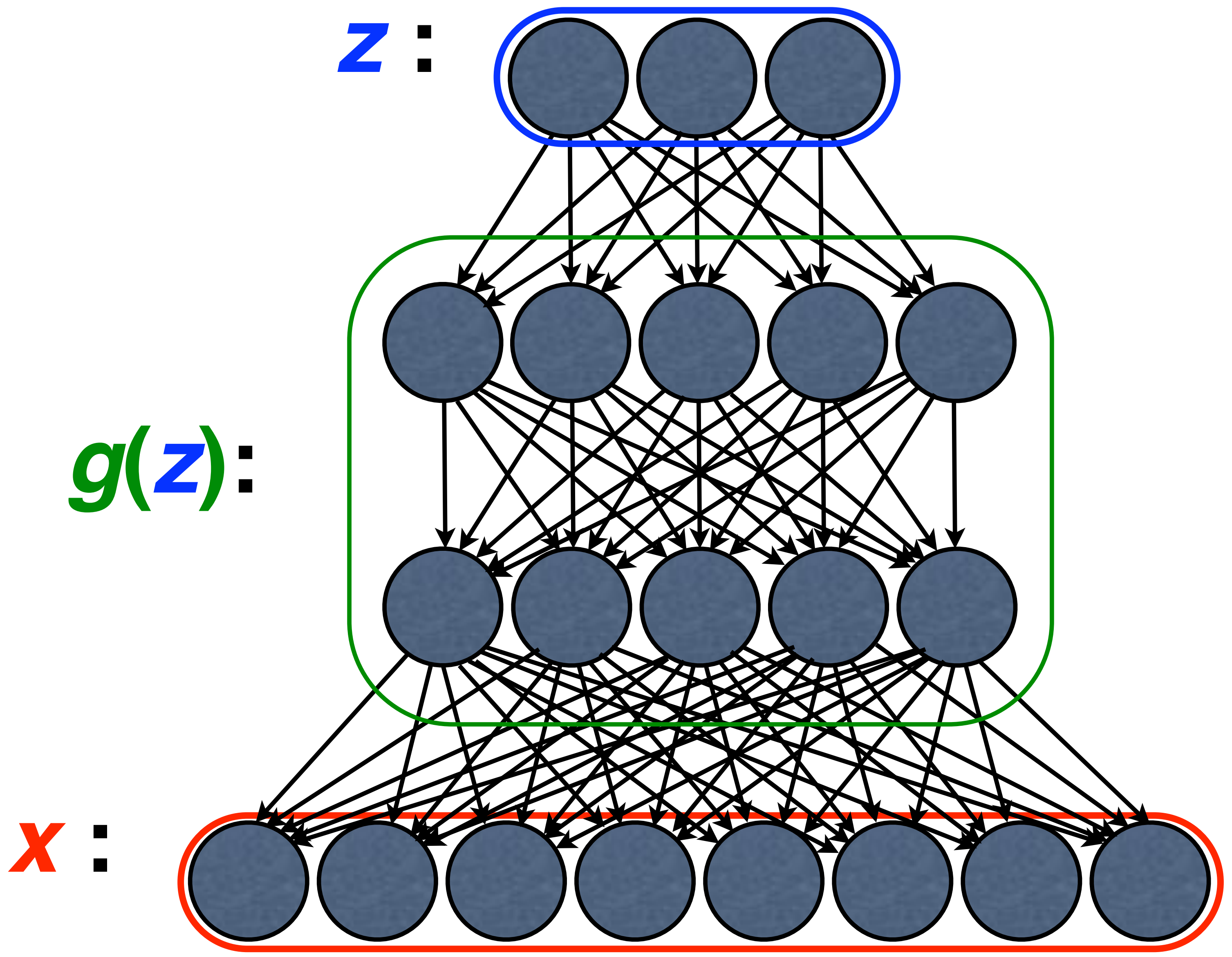


Image from: Ward, A. D., Hamarneh, G.: **3D Surface Parameterization Using Manifold Learning for Medial Shape Representation**, *Conference on Image Processing, Proc. of SPIE Medical Imaging*, 2007

Variational Auto-Encoder (VAE)

- Where does ***z*** come from? — The classic DAG problem.
- The VAE approach: introduce an inference machine $q_\phi(z \mid x)$ that learns to approximate the posterior $p_\theta(z \mid x)$.
- Define a variational lower bound on the data likelihood: $\log p_\theta(x) \geq \mathcal{L}(\theta, \phi, x)$

$$
\begin{aligned}
\log p_\theta(x) &\geq \log p_\theta(x) - D_{\mathrm{KL}}\left[q_\phi(z|x) \| p_\theta(z|x)\right] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x) + \log p_\theta(z|x) - \log q_\phi(z|x)\right] \\
&= \mathcal{L}(\theta, \phi, x) \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z) + \log p_\theta(z) - \log q_\phi(z|x)\right] \\
&= -D_{\mathrm{KL}}\left[q_\phi(z|x) \| p_\theta(z)\right] + \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right]
\end{aligned}
$$

- What is $q_\phi(z \mid x)$?

***regularization* term** ***reconstruction* term**

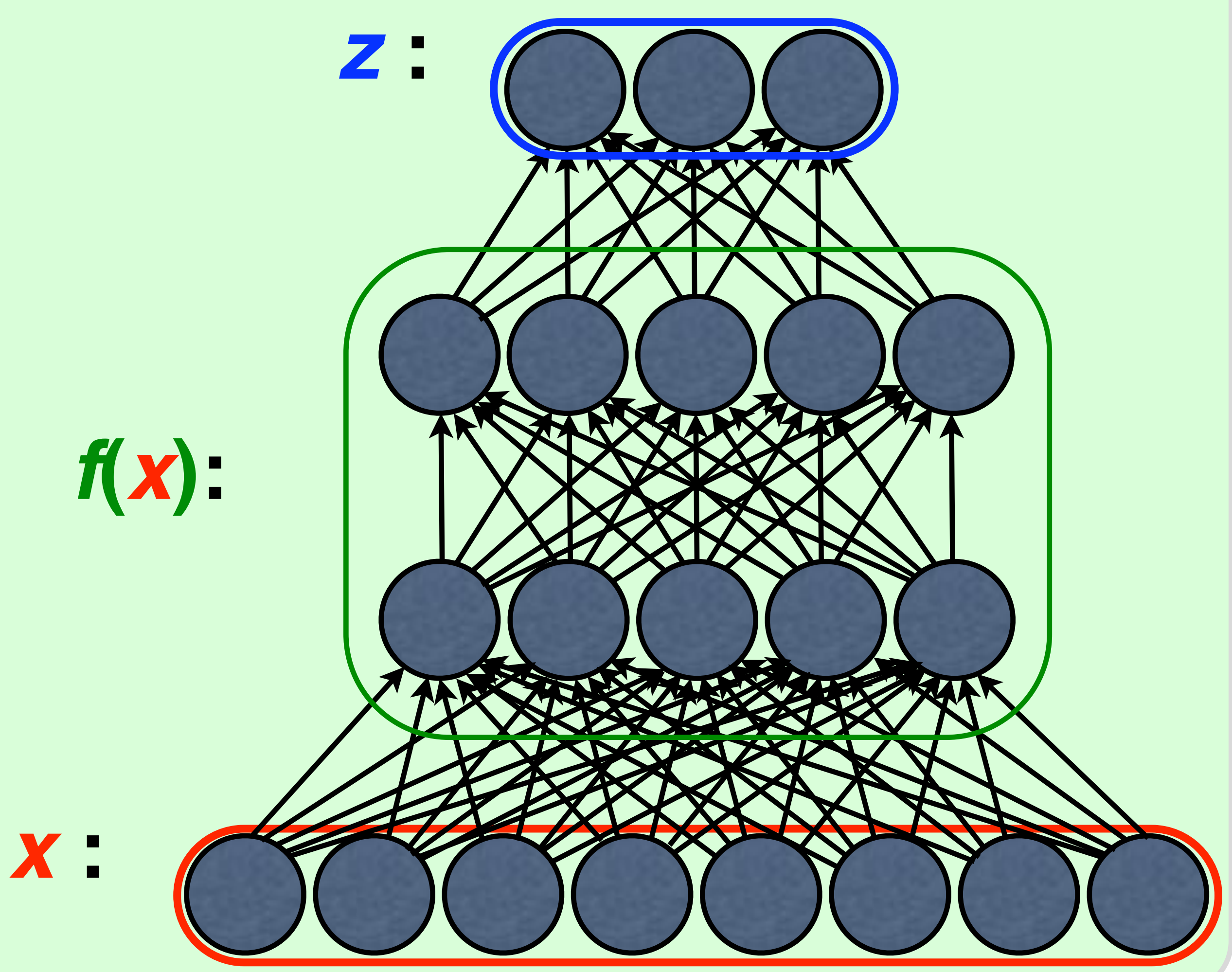
VAE Inference model

- The VAE approach: introduce an inference model $q_\phi(z \mid x)$ that learns to approximates the intractable posterior $p_\theta(z \mid x)$ by optimizing the variational lower bound:

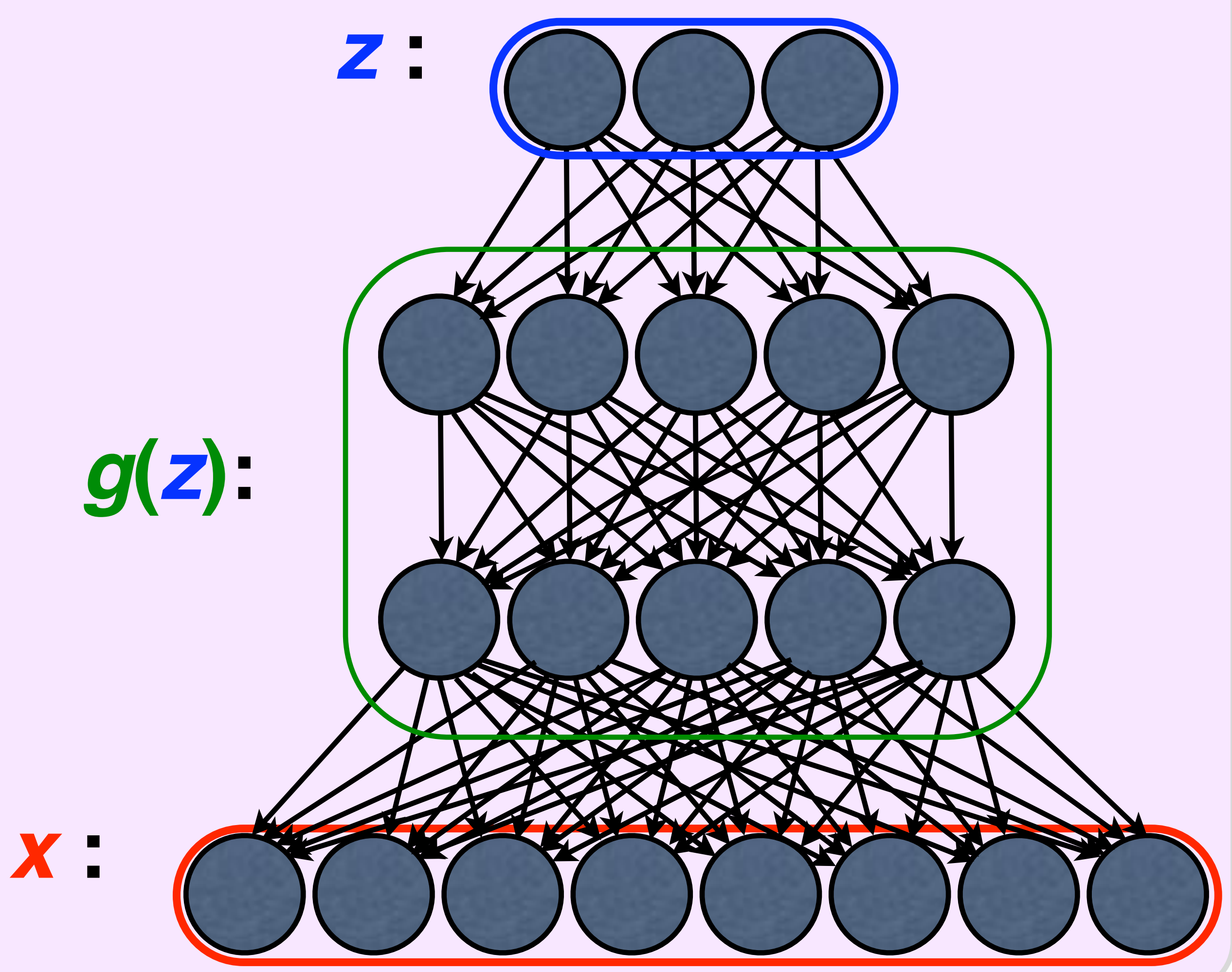
$$\mathcal{L}(\theta, \phi, x) = -D_{\mathrm{KL}}\left(q_\phi(z \mid x) \| p_\theta(z)\right) + \mathbb{E}_{q_\phi(z \mid x)}\left[\log p_\theta(x \mid z)\right]$$

- We parameterize $q_\phi(z \mid x)$ with another neural network:

$$q_\phi(z \mid x) = q(z; f(x, \phi))$$

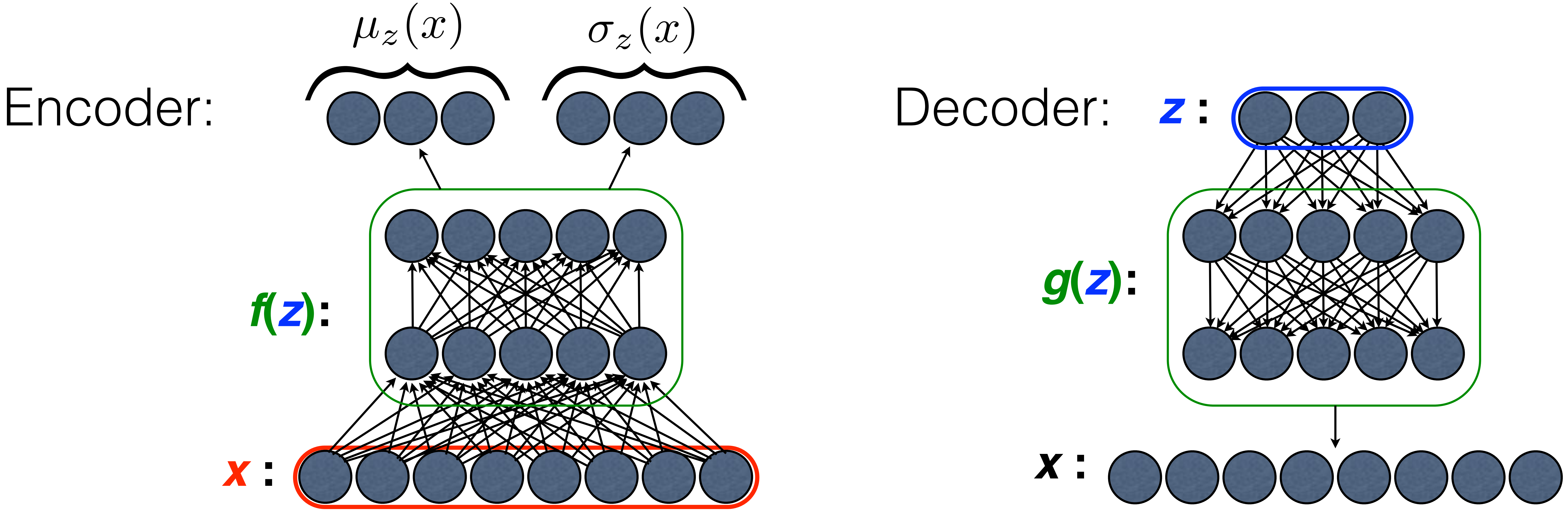


$$p_\theta(x \mid z) = p(x; g(z, \theta))$$



Reparametrization trick

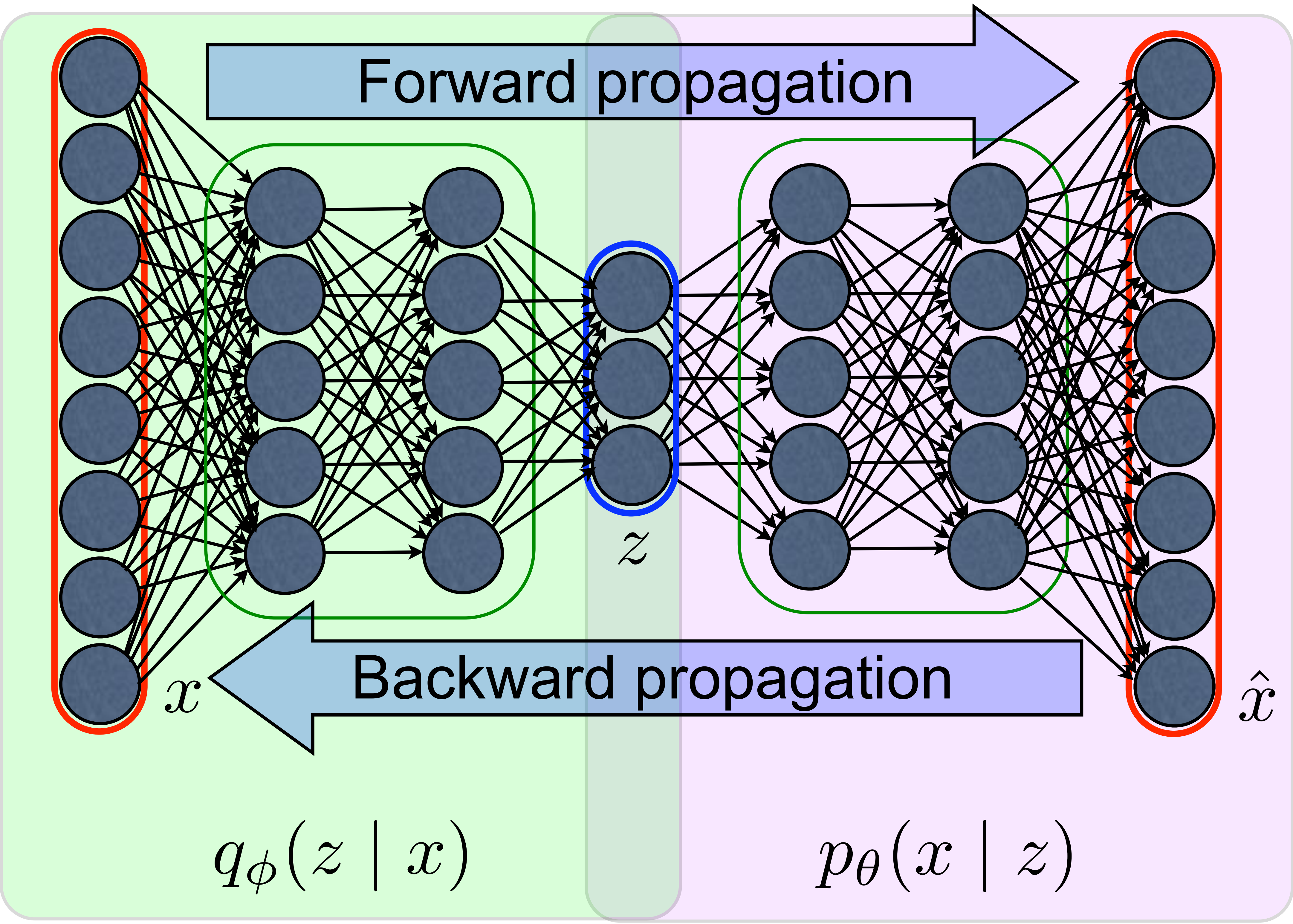
- Adding a few details + one really important trick
- Let's consider ***z*** to be real and $q_\phi(z \mid x) = \mathcal{N}(z; \mu_z(x), \sigma_z(x))$
- Parametrize ***z*** as $z = \mu_z(x) + \sigma_z(x)\epsilon_z$ where $\epsilon_z = \mathcal{N}(0, 1)$



Training with backpropagation!

- Due to a reparametrization trick, we can simultaneously train both the generative model $p_\theta(x \mid z)$ and the inference model $q_\phi(z \mid x)$ by optimizing the variational bound using gradient backpropagation.

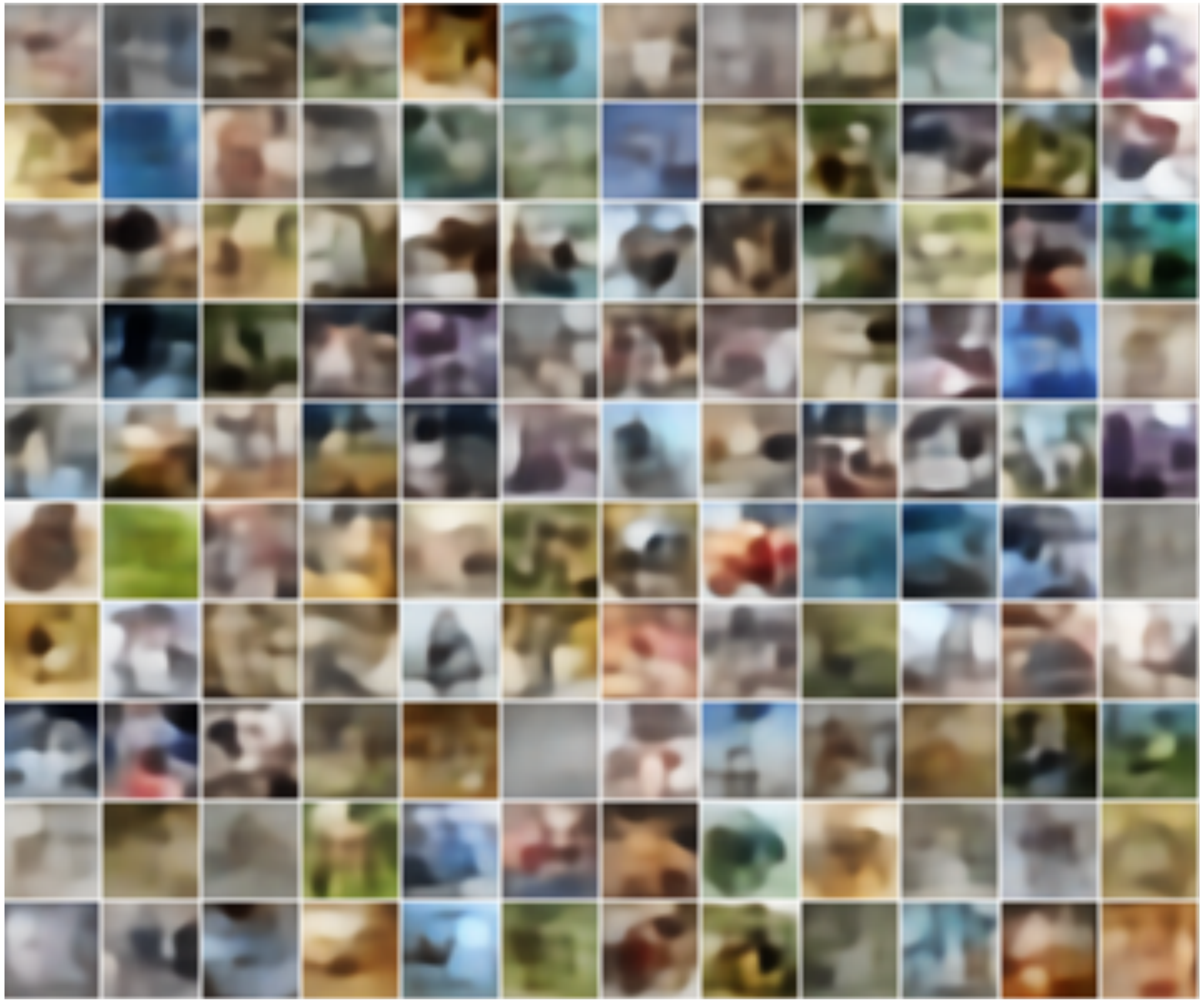
Objective function: $\mathcal{L}(\theta, \phi, x) = -D_{\mathrm{KL}}\left(q_\phi(z \mid x) \| \, p_\theta(z)\right) + \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x \mid z)\right]$



vanilla VAE samples

Impressive …
… at the time

Labelled Faces in the Wild (LFW)



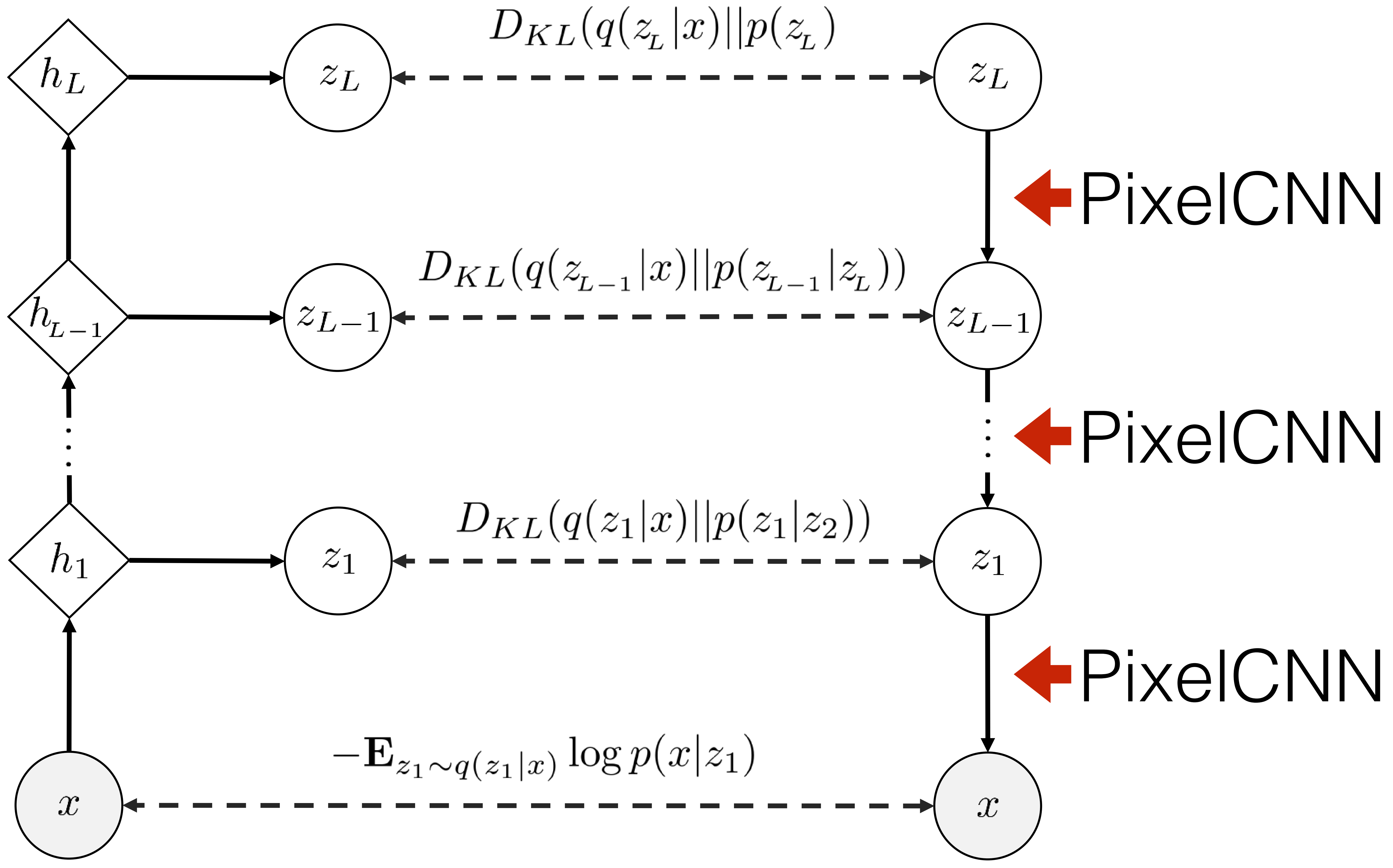
ImageNet (small)

PixelVAE

Ishaan Gulrajani, Kundan Kumar, Faruk Ahmed Adrien Ali Taiga, Francesco Visin, David Vazquez, Aaron Courville. ICLR 2017

MILA

- Uses a PixelCNN in the VAE decoder to help avoid the blurring caused by the standard VAE assumption of independent pixels.



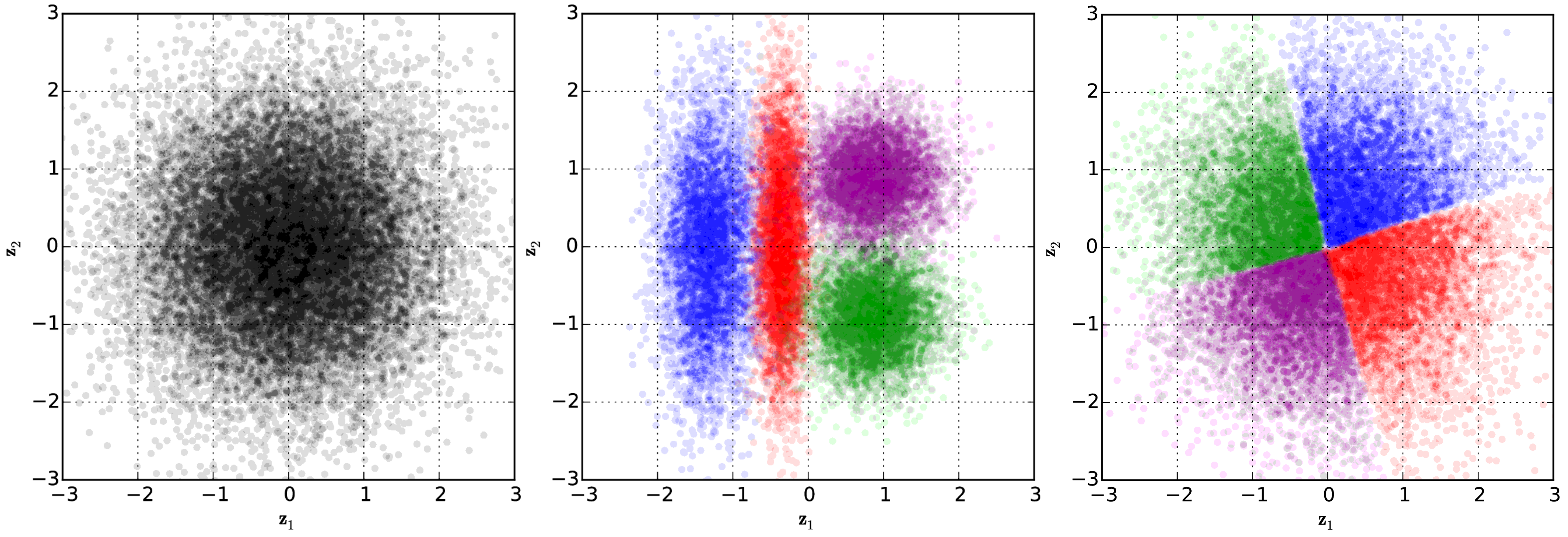
PixelVAE Samples (Gulrajani et al. 2017)

MILA

LSUN bedroom scenes (64x64)

ImageNet (64x64)

Inverse Autoregressive Flow (Kingma et al., NIPS 2016)



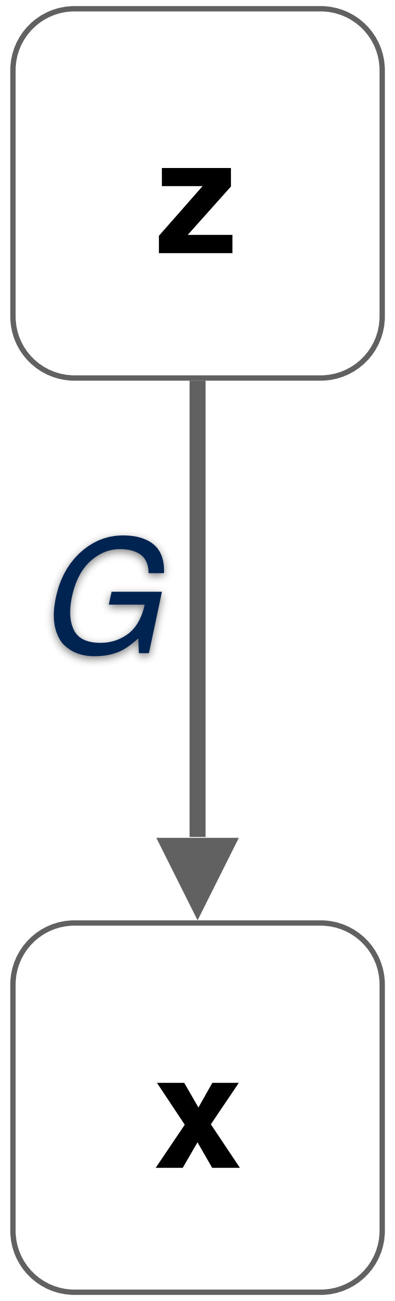
(a) Prior distribution (b) Posteriors in standard VAE (c) Posteriors in VAE with IAF

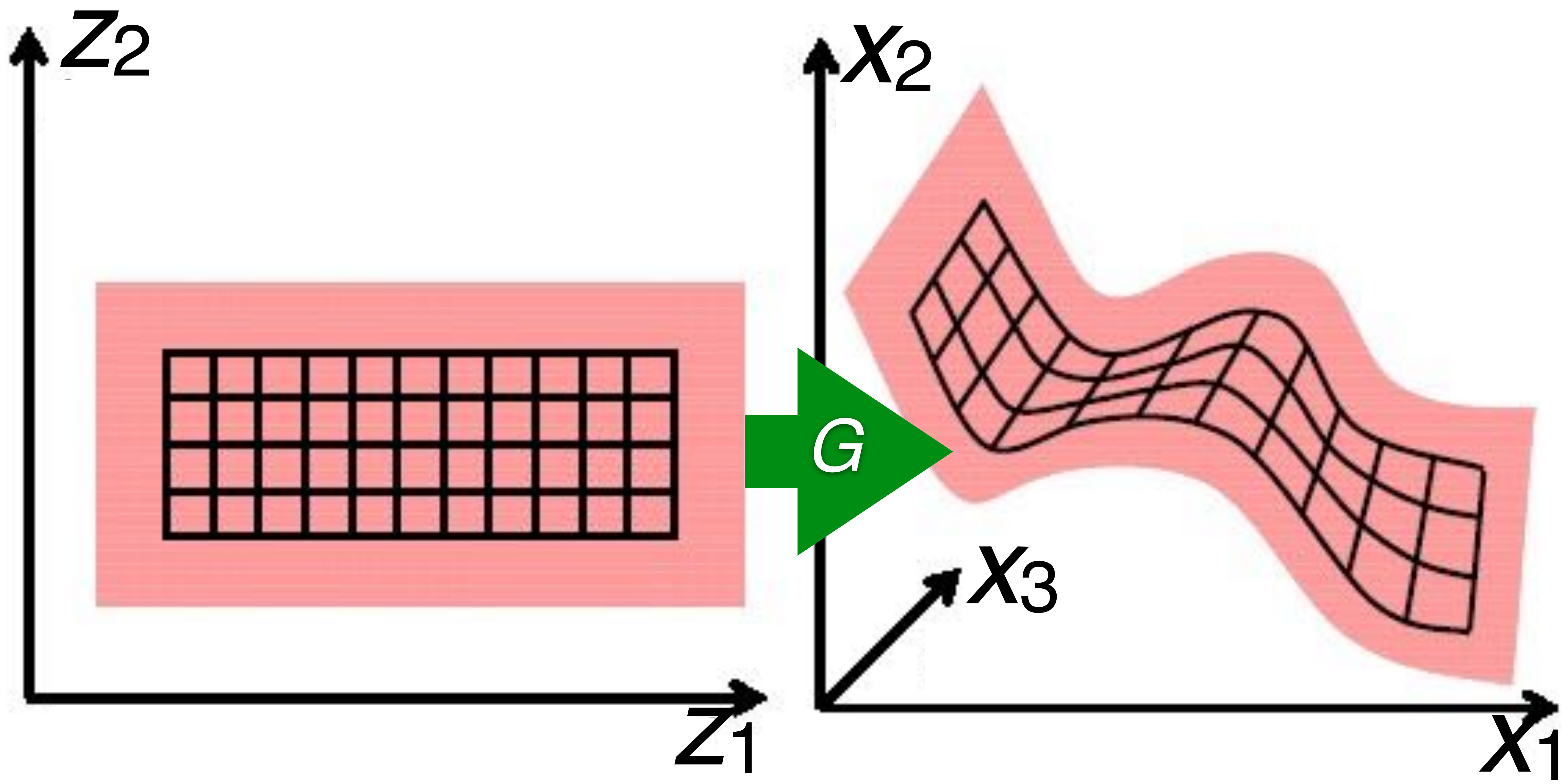
- Standard VAE posteriors are factorized - limiting how well they can (marginally) fit the prior.
- IAF greatly improves the flexibility of the posterior distributions, and allows for a much better fit between the posteriors and the prior.

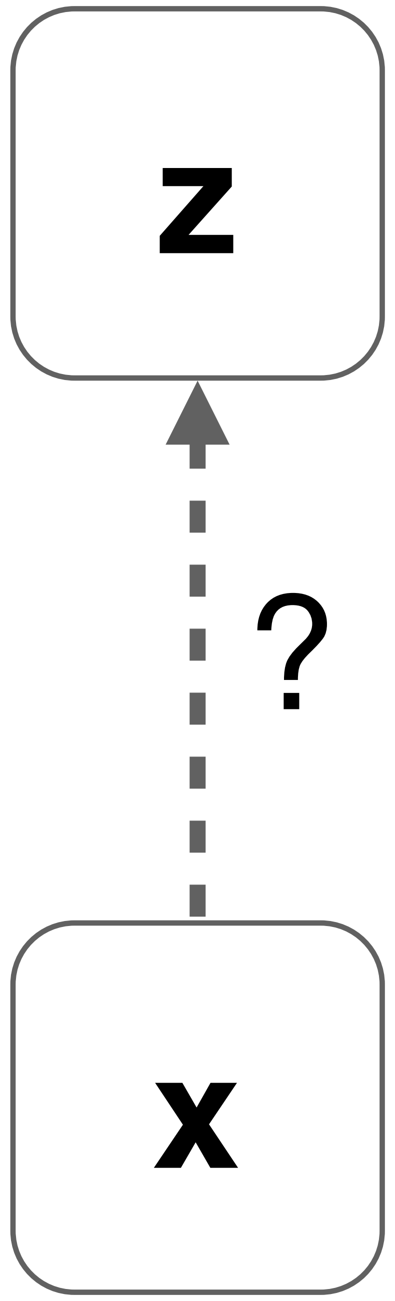
Another way to train a latent variable model?

inference

Latent variables

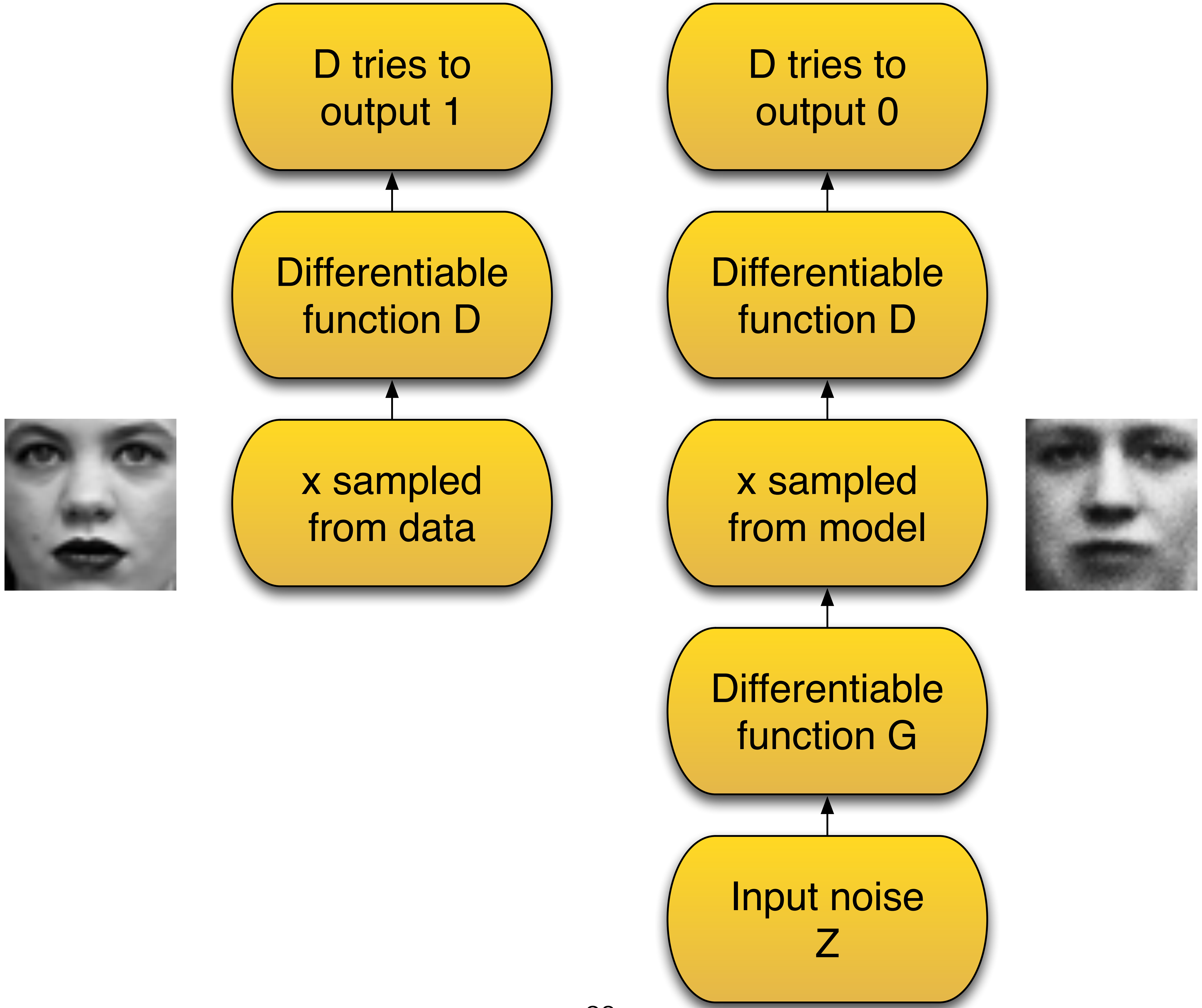




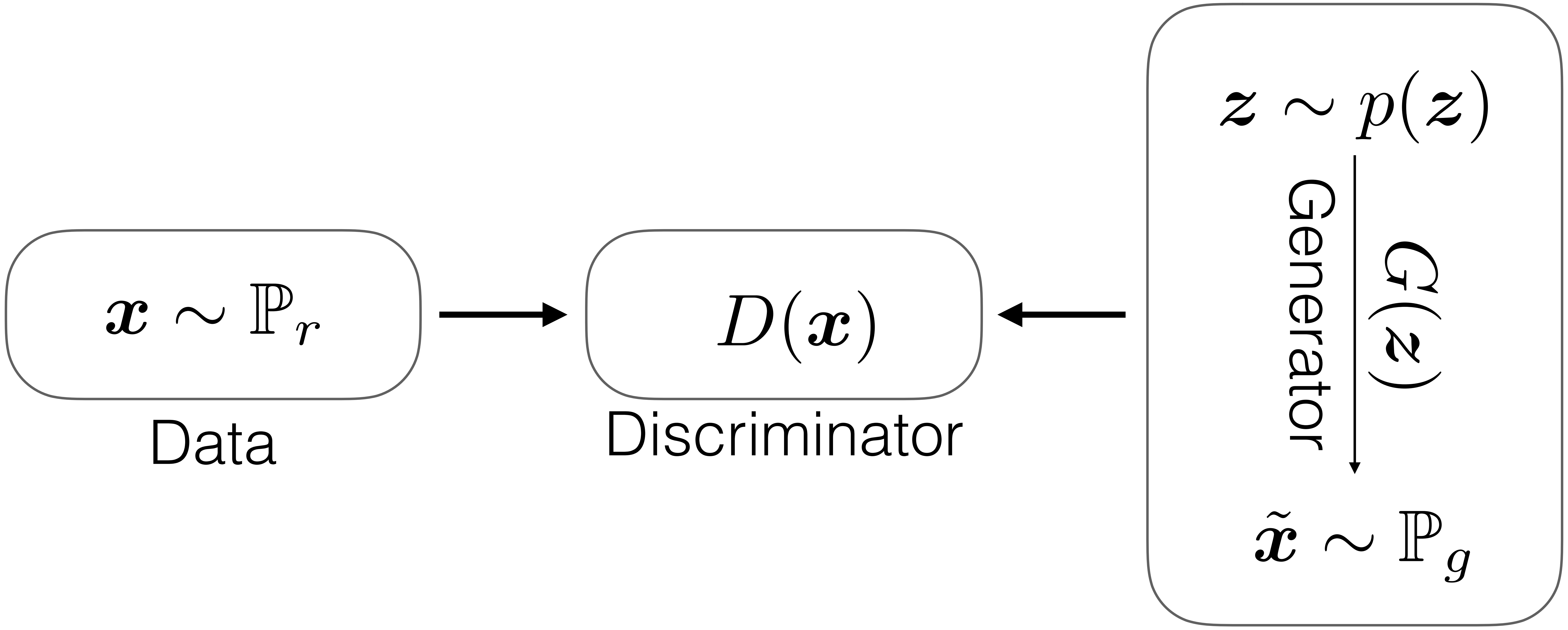


Observed variables

Generative Adversarial Networks



Generative Adversarial Networks



GAN Objective

- Formally, express the game between discriminator D and generator G with the minimax objective:

$$\min_{G} \max_{D} \mathop{\mathbb{E}}_{\boldsymbol{x} \sim \mathbb{P}_r} [\log(D(\boldsymbol{x}))] + \mathop{\mathbb{E}}_{\tilde{\boldsymbol{x}} \sim \mathbb{P}_g} [\log(1 - D(\tilde{\boldsymbol{x}}))].$$

where:

- $\mathbb{P}_r$ is the data distribution
- $\mathbb{P}_g$ is the model distribution implicitly defined by:

$$\tilde{\boldsymbol{x}} = G(\boldsymbol{z}), \quad \boldsymbol{z} \sim p(\boldsymbol{z})$$

- the generator input $\boldsymbol{z}$ is sampled from some simple noise distribution, (e.g. uniform or Gaussian).

GAN Theory

- Optimal (nonparametric) discriminator:

$$D^*(\boldsymbol{x}) = \frac{p_r(\boldsymbol{x})}{p_r(\boldsymbol{x}) + p_g(\boldsymbol{x})}$$

- Under an ideal discriminator, the generator minimizes the Jensen-Shannon divergence between $\mathbb{P}_r$ and $\mathbb{P}_g$.

$$\mathrm{JS}(\mathbb{P}_r \| \mathbb{P}_g) = \mathrm{KL}\left(\mathbb{P}_r \middle\| \frac{\mathbb{P}_r + \mathbb{P}_g}{2}\right) + \mathrm{KL}\left(\mathbb{P}_g \middle\| \frac{\mathbb{P}_r + \mathbb{P}_g}{2}\right)$$

where $\mathrm{KL}(\mathbb{P}_r \| \mathbb{P}_g) = \int \log\left(\frac{p_r(x)}{p_g(x)}\right) p_r(x) d\mu(x)$

GAN Theory … in practice

- The minimax objective leads to vanishing gradients as the discriminator saturates.

- In practice, Goodfellow et al (2014) advocate the heuristic training objective:

$$\max_{D} \underset{\boldsymbol{x}\sim\mathbb{P}_r}{\mathbb{E}}[\log(D(\boldsymbol{x}))] + \underset{\tilde{\boldsymbol{x}}\sim\mathbb{P}_g}{\mathbb{E}}[\log(1 - D(\tilde{\boldsymbol{x}}))].$$

$$\max_{G} \underset{\tilde{\boldsymbol{x}}\sim\mathbb{P}_g}{\mathbb{E}}[\log(D(\tilde{\boldsymbol{x}}))].$$

- However, this modified loss function can still misbehave in the presence of a good discriminator.

GAN samples

Least-Squares GAN

Xudong Mao, Qing Li†, Haoran Xie, Raymond Y.K. Lau and Zhen Wang, ArXiv, Feb. 2017

MILA

128x128 LSUN bedroom scenes

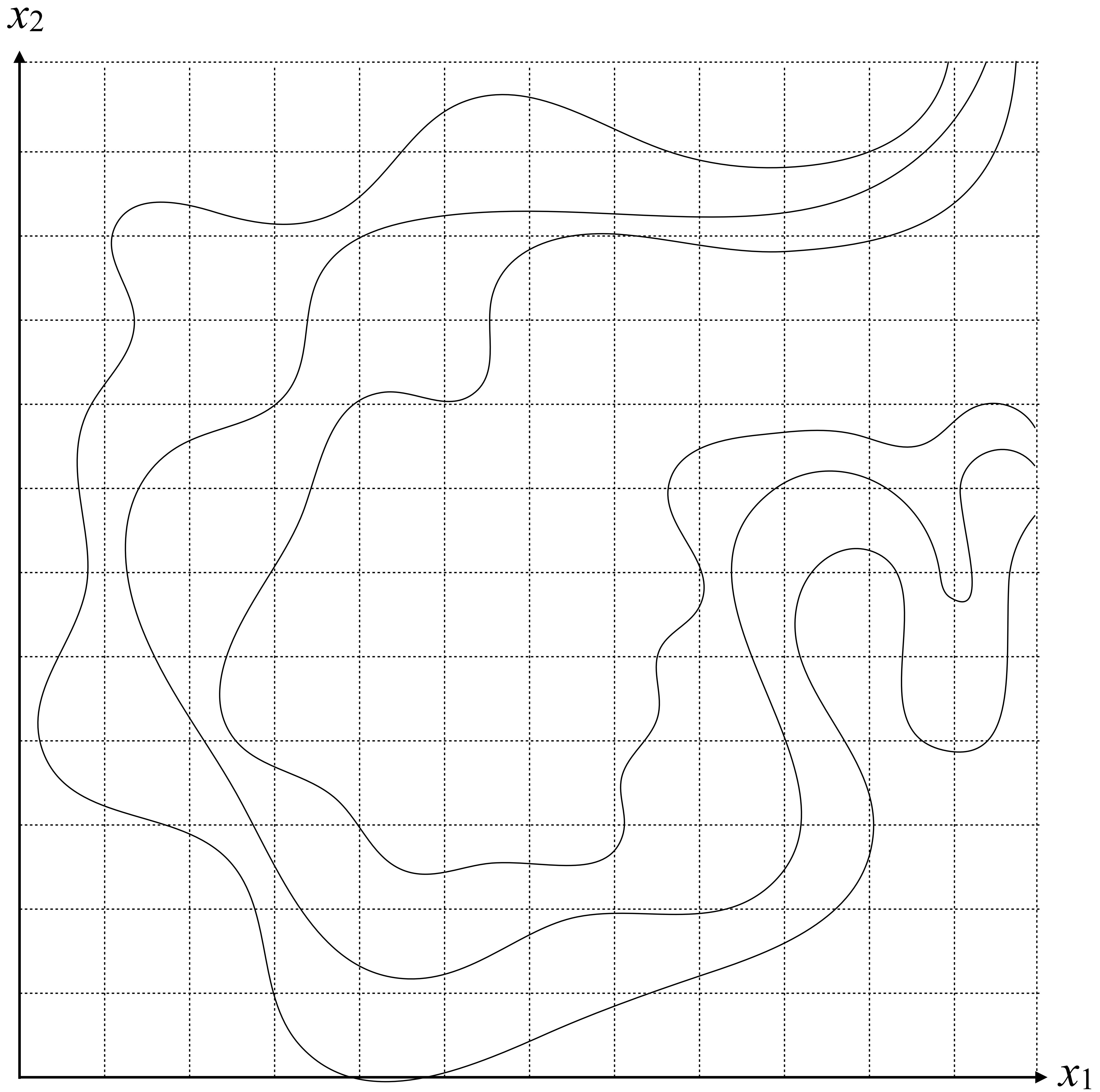
DCGAN samples (Radford, Metz and Chintala; 2016)

Z-space interpolations

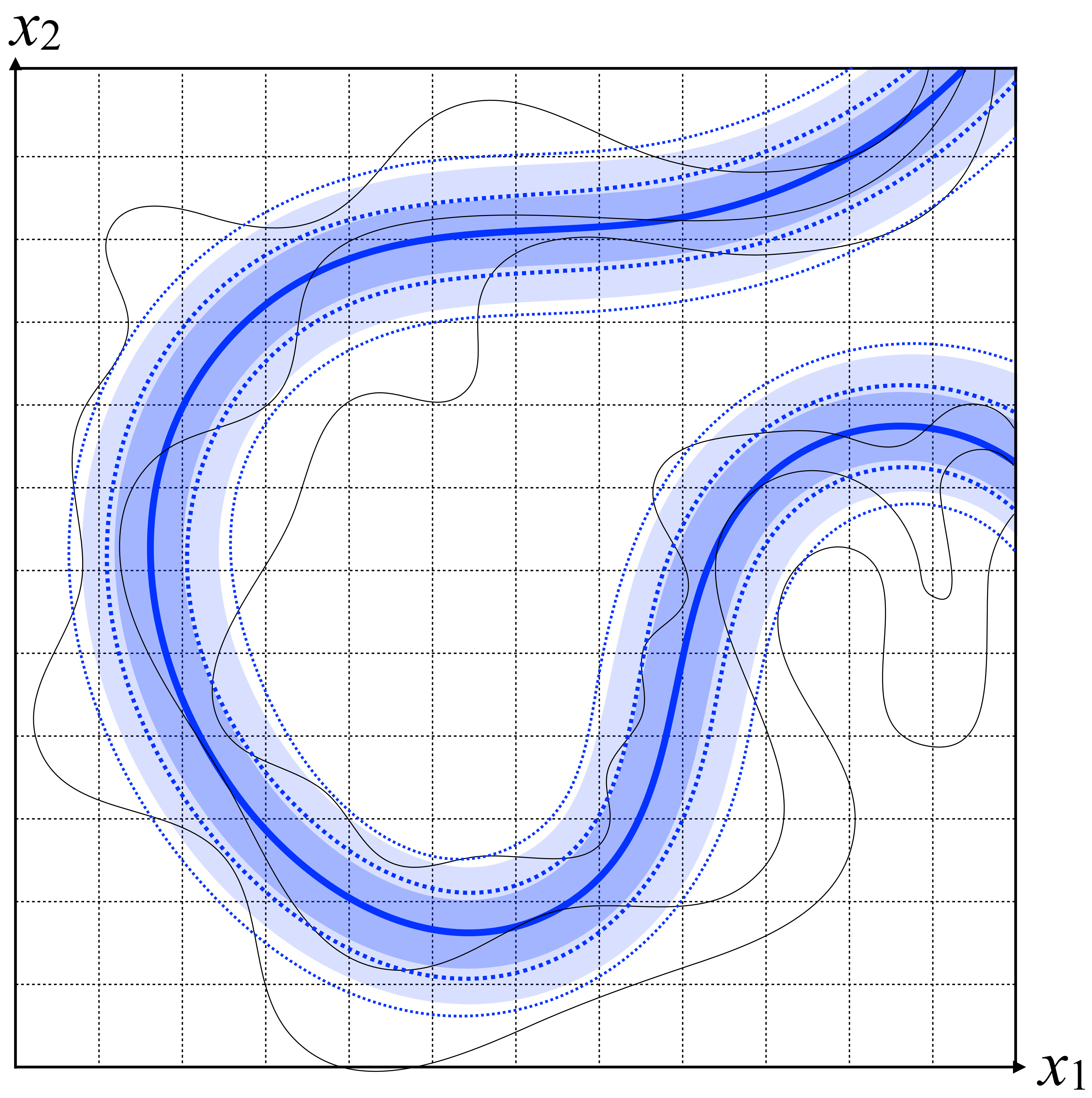
LSUN bedroom scenes

What makes GANs special?

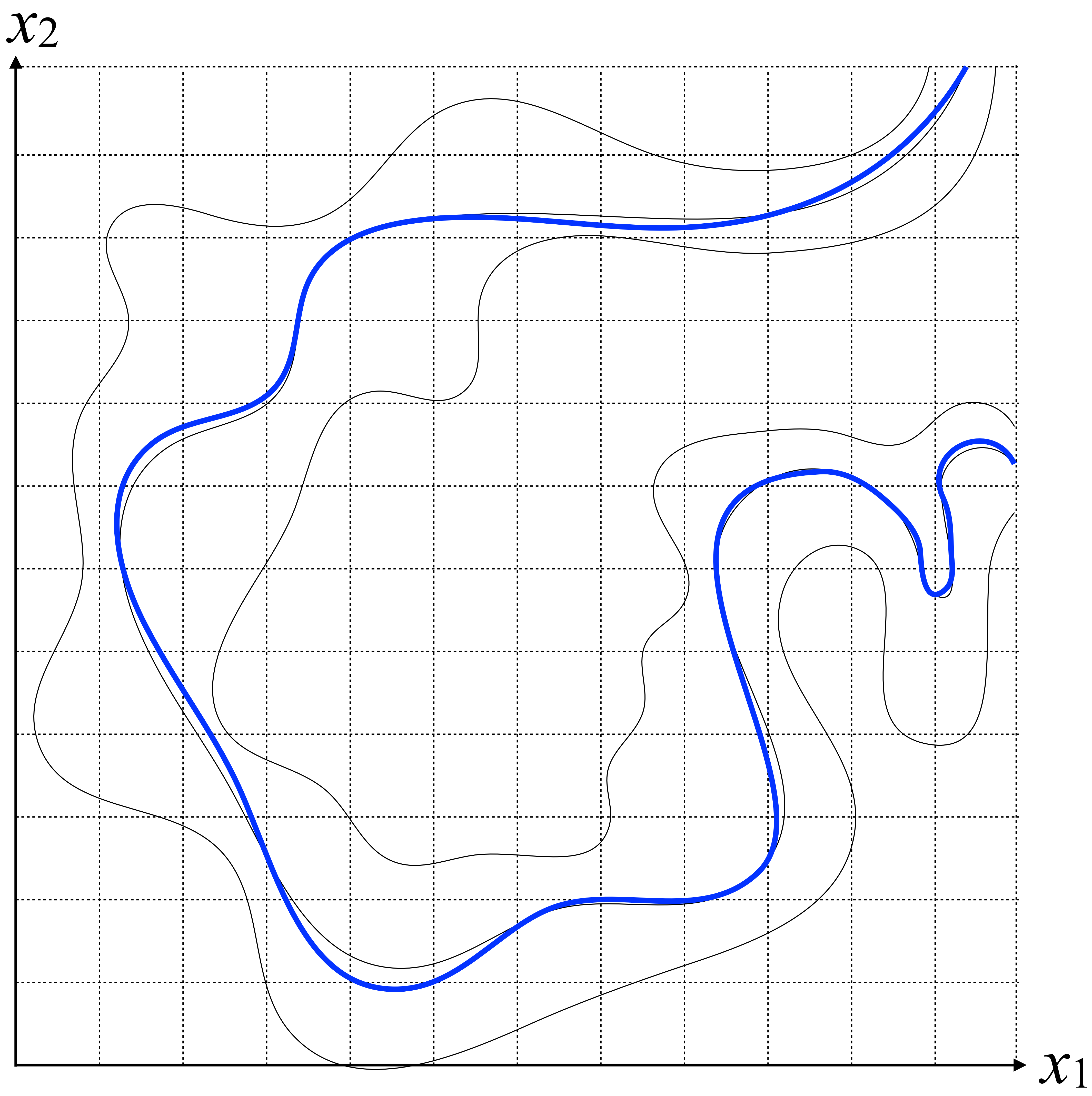
Cartoon of the Image manifold:



What makes GANs special?



more traditional max-likelihood approach

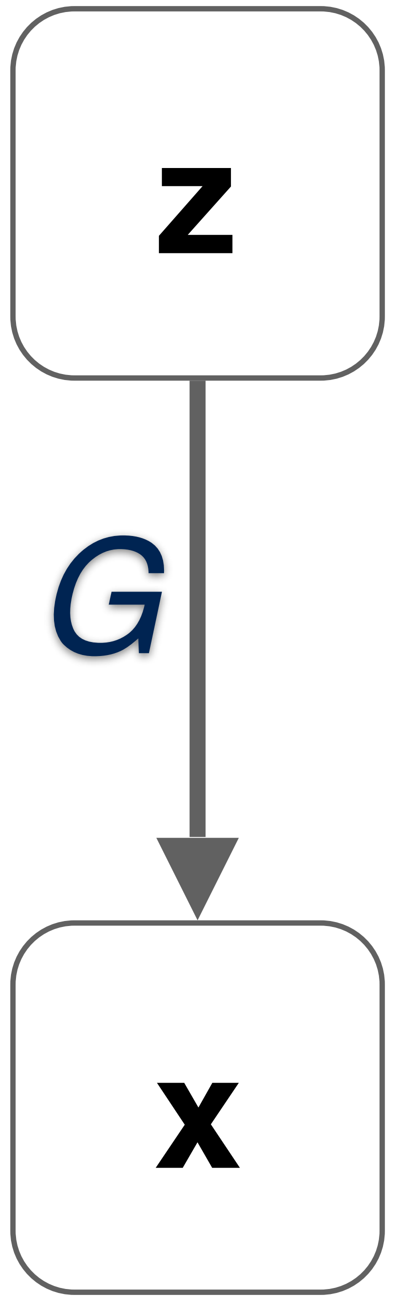


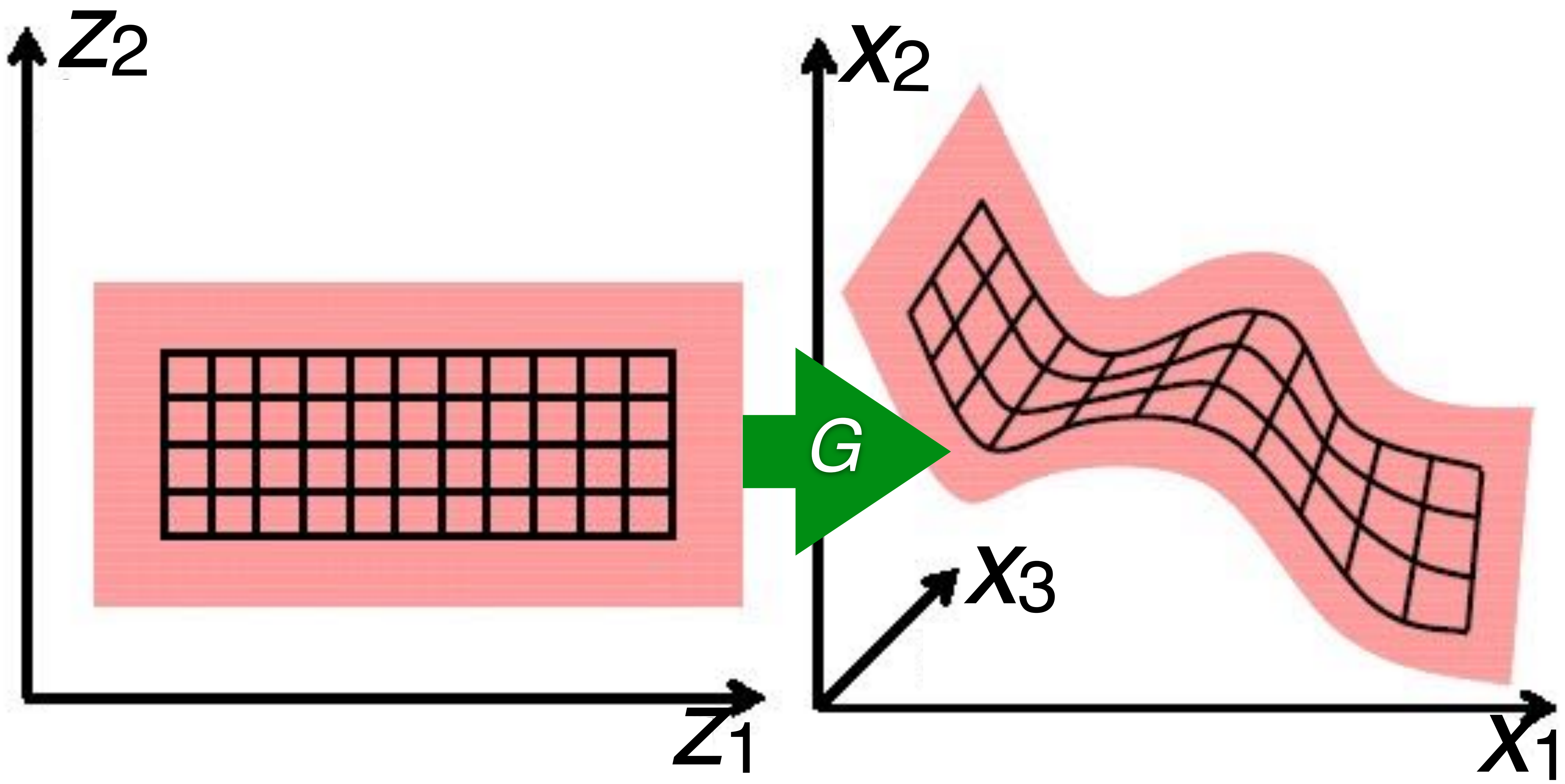
GAN

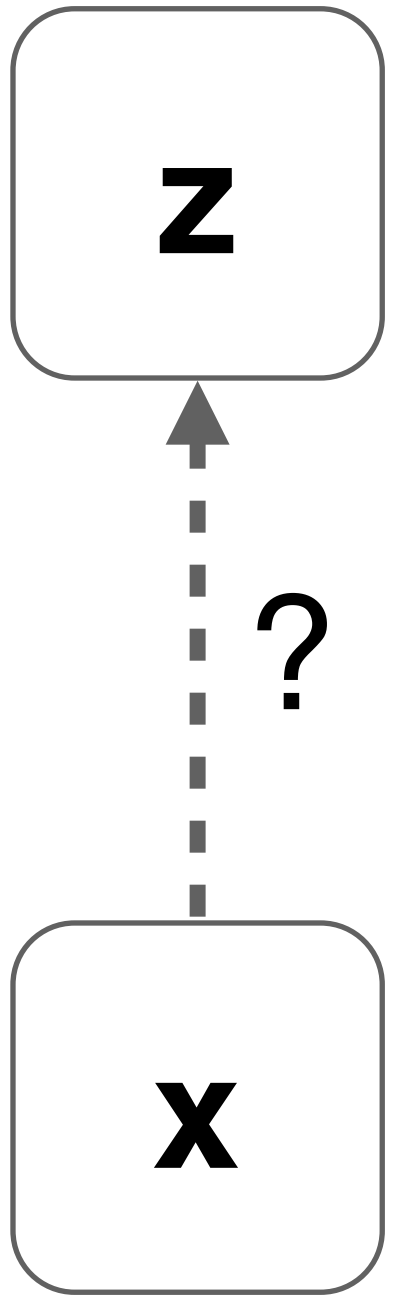
But what about inference...

- Can we incorporate an inference mechanism into GANs?

Latent variables

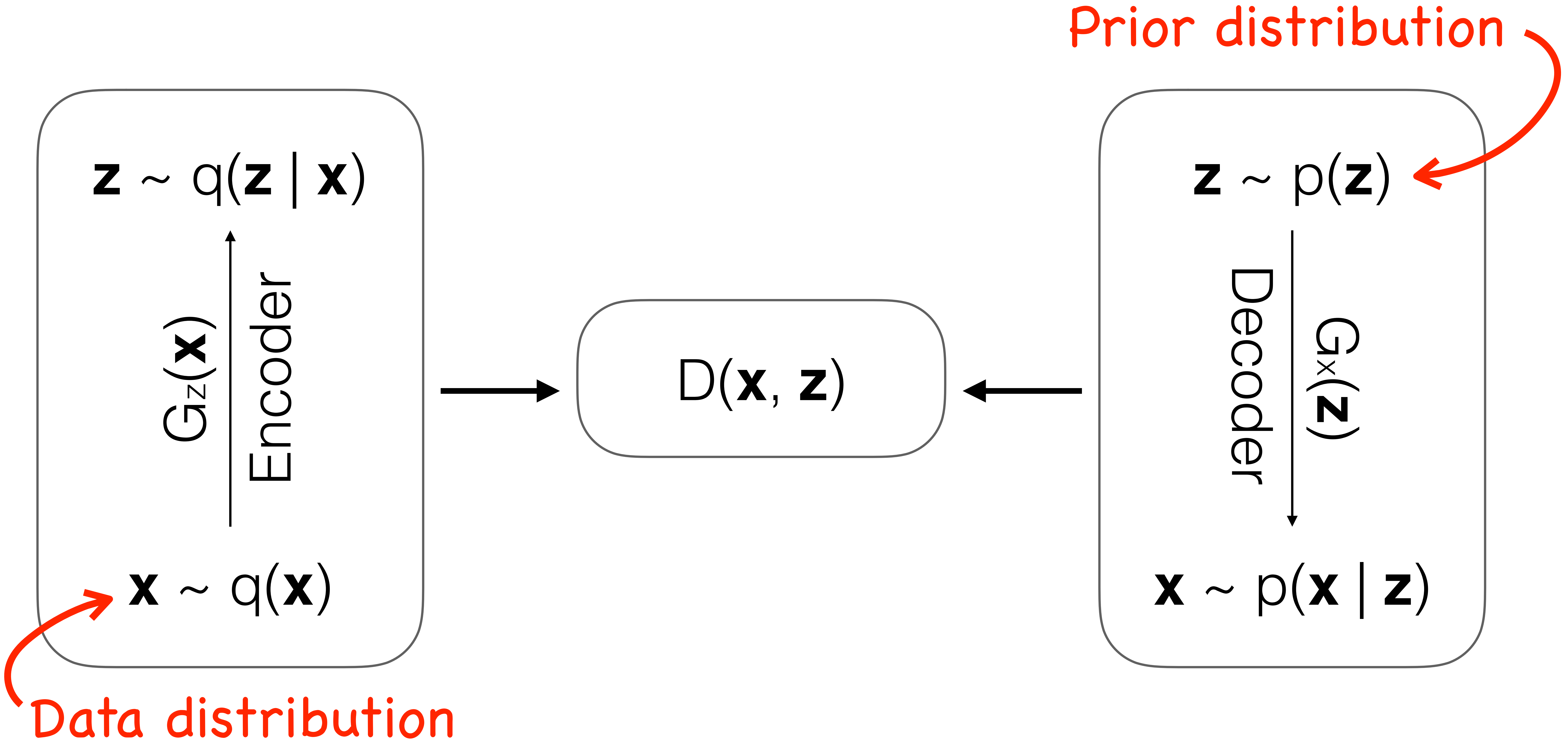






Observed variables

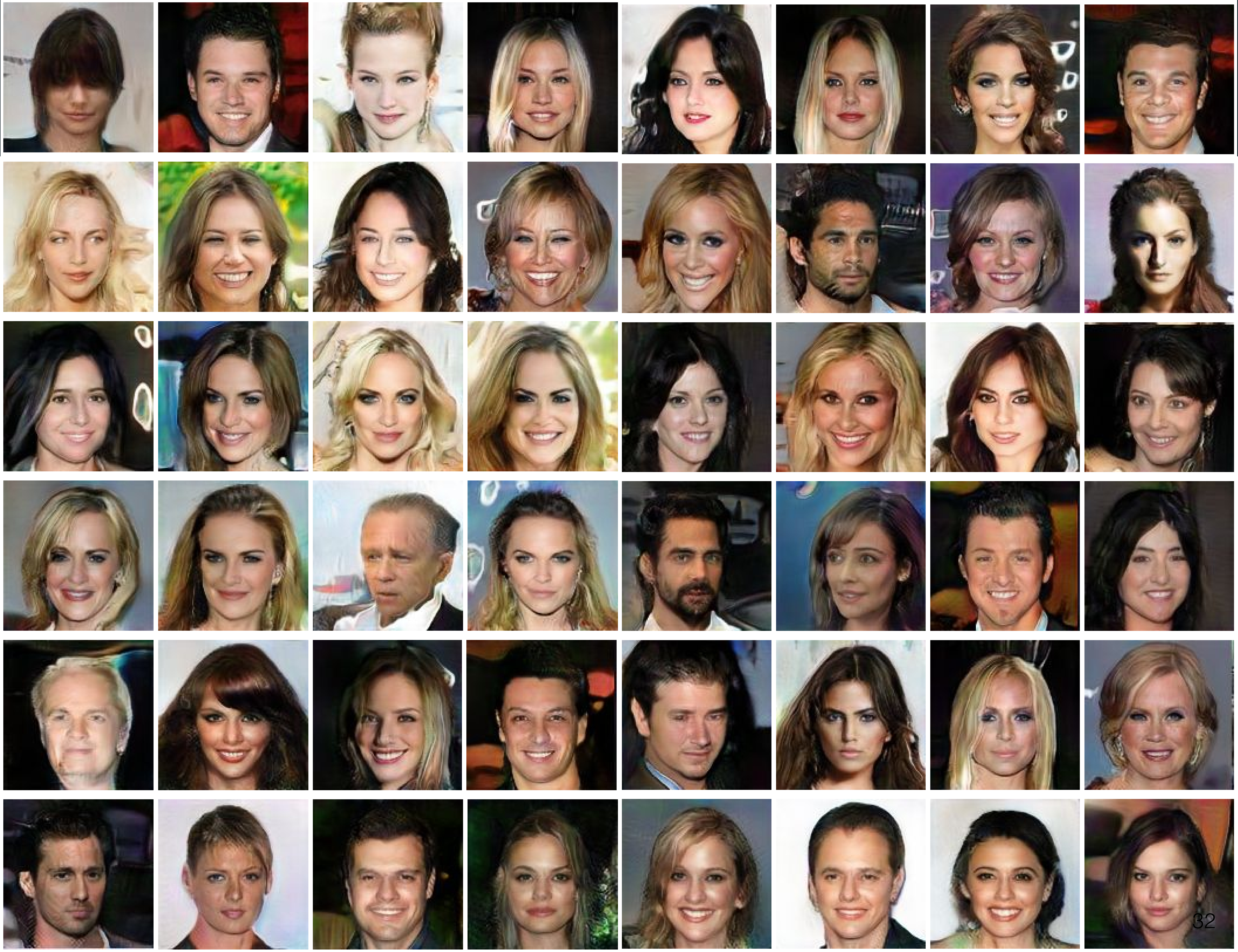
ALI / BiGAN: model diagram



- **ALI**: Vincent Dumoulin, Ishmael Belghazi, Olivier Mastropietro, Ben Poole, Alex Lamb, Martin Arjovsky (2016) *ADVERSARIALLY LEARNED INFERENCE*, arXiv:1606.00704, ICLR 2017
- **BiGAN**: Donahue, Krähenbühl and Darrell (2016), *ADVERSARIAL FEATURE LEARNING*, arXiv:1605.09782, ICLR 2017

Hierarchical ALI

CelebA-128X128



Model samples

Hierarchical ALI: CelebA-128X128

Reconstructions given z_1, z_2

Reconstructions given z_2

cycleGAN: Adversarial training of domain transformations

(Zhu et al. ICCV 2017)

- CycleGAN learns transformations across domains with unpaired data.
- Combines GAN loss with "cycle-consistency loss": L1 reconstruction.

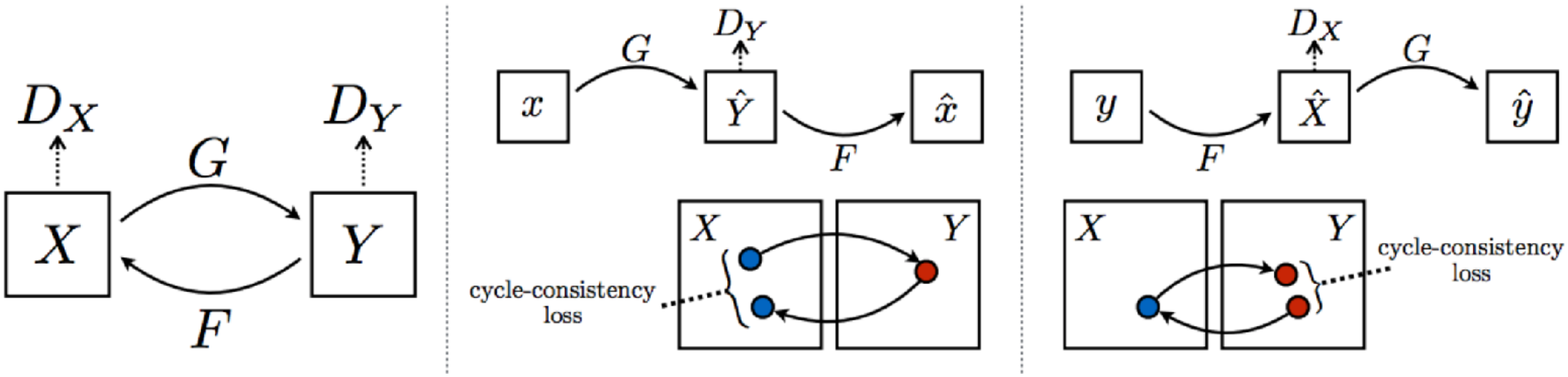


Image credits: Jun-Yan Zhu*, Taesung Park*, Phillip Isola, and Alexei A. Efros. "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", in IEEE International Conference on Computer Vision (ICCV), 2017.

CycleGAN for unpaired data

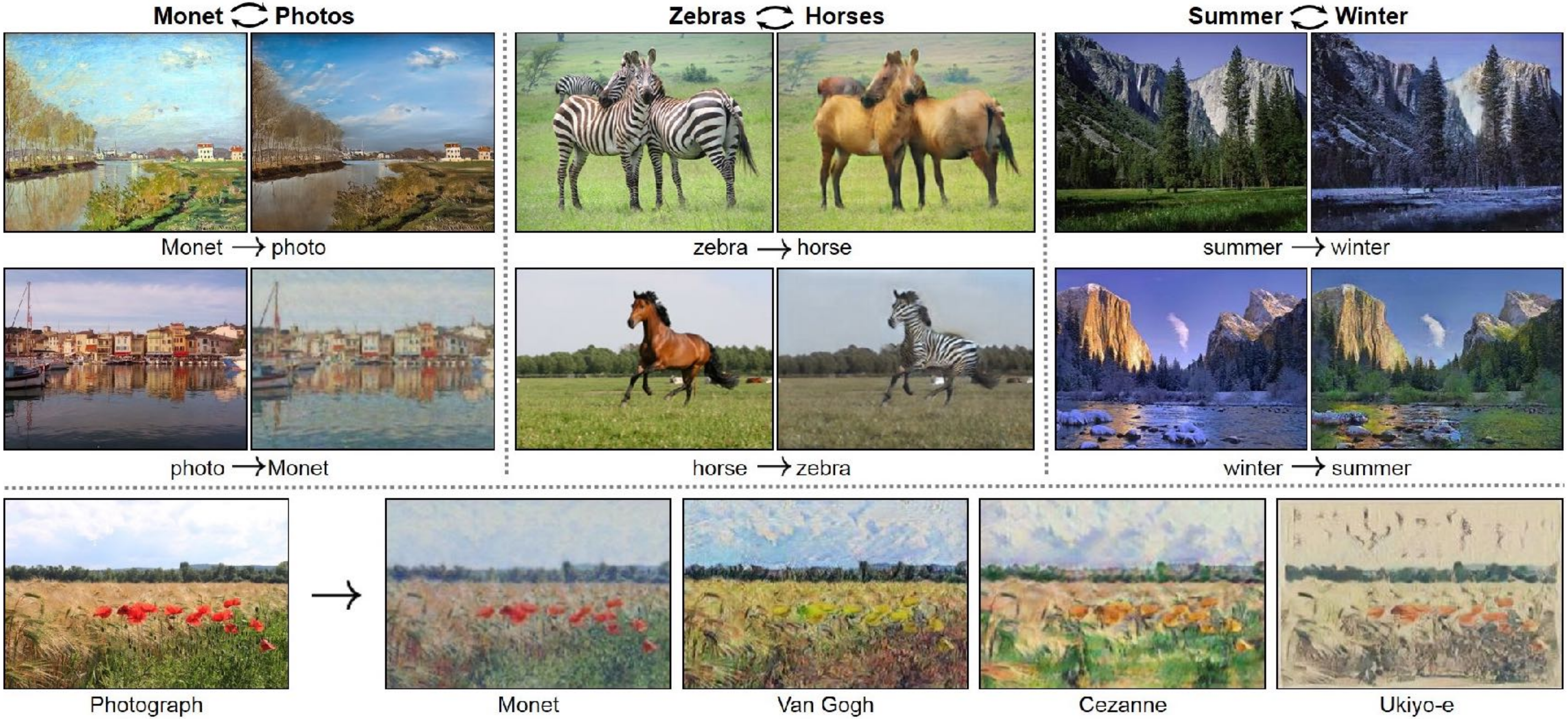


Image credits: Jun-Yan Zhu*, Taesung Park*, Phillip Isola, and Alexei A. Efros. "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", in IEEE International Conference on Computer Vision (ICCV), 2017.

PROGRESSIVE GROWING OF GANS FOR IMPROVED QUALITY, STABILITY, AND VARIATION (Kerras et al. from NVIDIA, 2017)

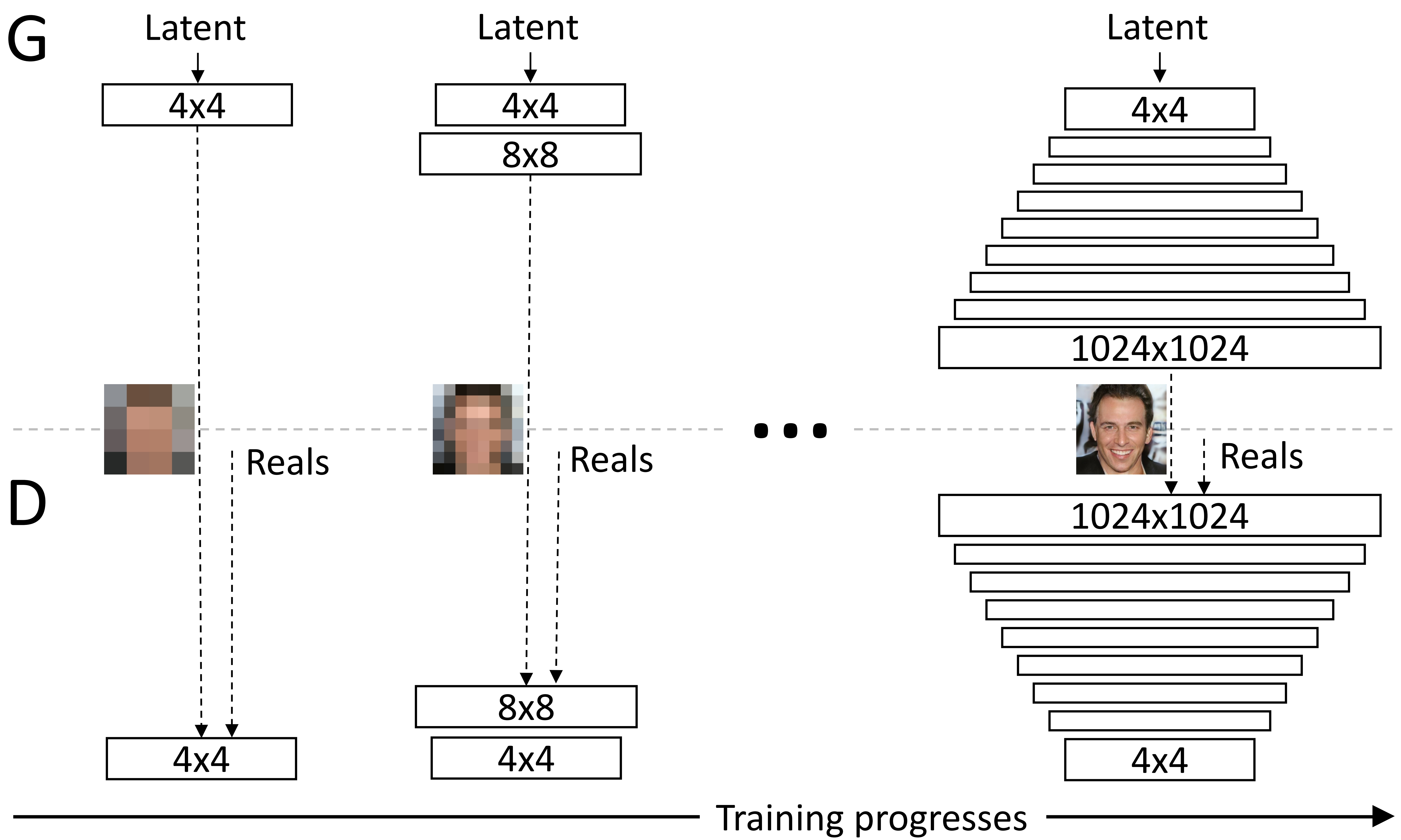
MILA

- Recent work from NVIDIA.
- Improves image quality by growing the model size throughout training.
- Samples from a model trained on the CelebA face dataset.

1024x1024 model samples

PROGRESSIVE GROWING OF GANS FOR IMPROVED QUALITY, STABILITY, AND VARIATION (Kerras et al. from NVIDIA, 2017)

MILA



PROGRESSIVE GROWING OF GANS FOR IMPROVED QUALITY, STABILITY, AND VARIATION (Kerras et al. from NVIDIA, 2017)

MILA

- Recent work from NVIDIA.
- Improves image quality by growing the model size throughout training.
- Conditional samples from a model trained on the LSUN dataset

Model samples