CASE SCHOOL OF ENGINEERING
CASE WESTERN RESERVE UNIVERSITY

CSDS 600: Deep Generative Models

Variational Autoencoder (2)

Yu Yin (yu.yin@case.edu)

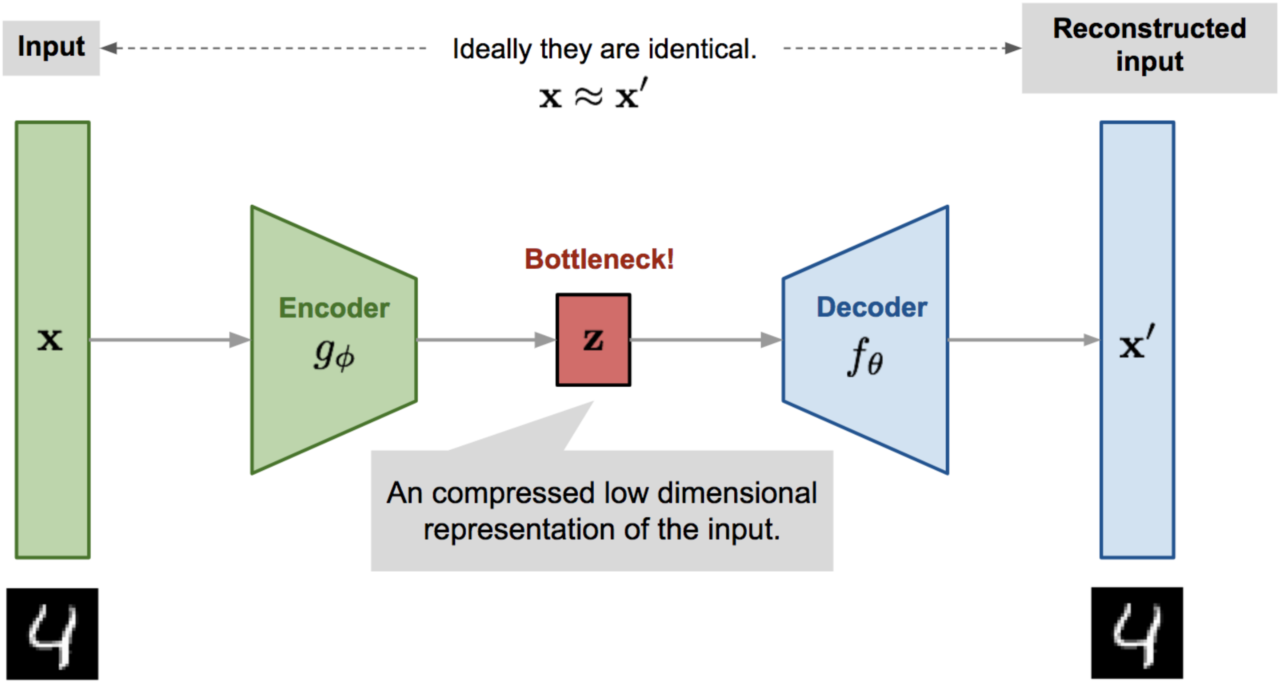
Case Western Reserve University

https://yin-yu.github.io/

Recap: Vanilla Autoencoder

What is it?

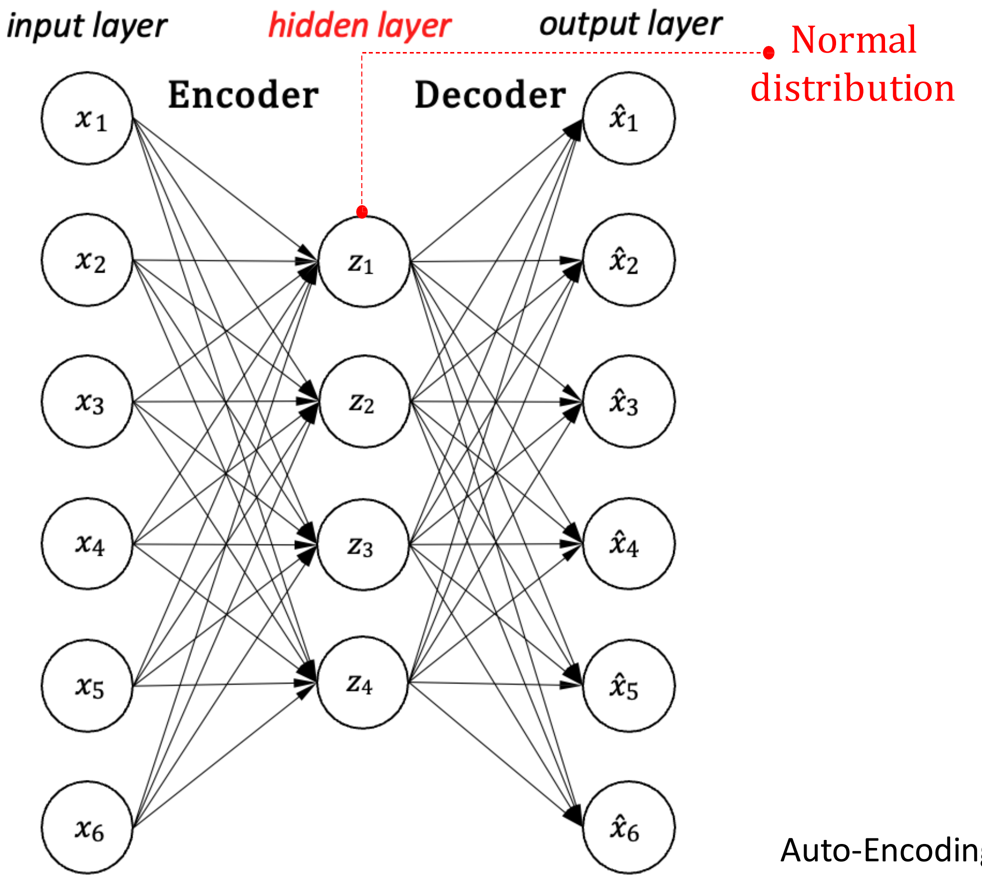
- Reconstruct high-dimensional data using a neural network model with a narrow bottleneck layer.
- It consists of two networks:
    - Encoder network: translates the original high-dimension input into the latent low-dimensional code.
    - Decoder network: recovers the data from the code



Weng, Lilian, From Autoencoder to Beta-VAE, 2018

Recap: VAE

- How to perform generation (sampling)?
- Instead of mapping the input into a fixed vector, we want to map it into a distribution p_θ, *e.g.*, Normal distribution



Auto-Encoding Variational Bayes. Diederik P. Kingma, Max Welling. ICLR 2013

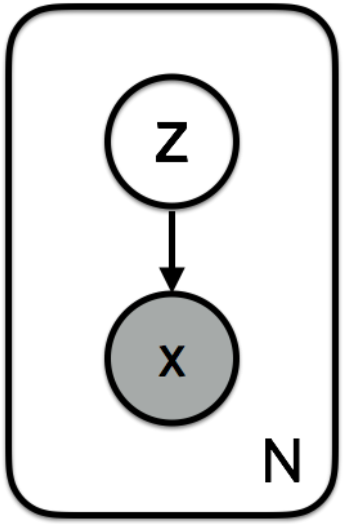
Outline

- Vanilla Autoencoder (AE)
- Denoising Autoencoder
- Sparse Autoencoder
- Contractive Autoencoder
- Stacked Autoencoder
- Variational Autoencoder (VAE)
  - From Neural Network Perspective
  - From Probability Model Perspective
- Convolutional VAE
- Conditional VAE

VAE: Variational Autoencoder

From Probability Model Perspective

- Instead of mapping the input into a **fixed** vector, we want to map it into a **distribution** p_θ, *e.g.*, Normal distribution
- The generative process can be written as follows:
  - $\mathbf{z}^{(i)} \sim p_{\theta^*}(\mathbf{z})$
  - $\mathbf{x}^{(i)} \sim p_{\theta^*}(\mathbf{x}|\mathbf{z}=\mathbf{z}^{(i)})$



Auto-Encoding Variational Bayes. Diederik P. Kingma, Max Welling. ICLR 2013

VAE: Variational Autoencoder

- Suppose that our joint distribution is $p_\theta(\mathbf{x}, \mathbf{z})$.
- Given $\mathcal{D} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(n)}\}$, maximizing the probability of generating real data samples:

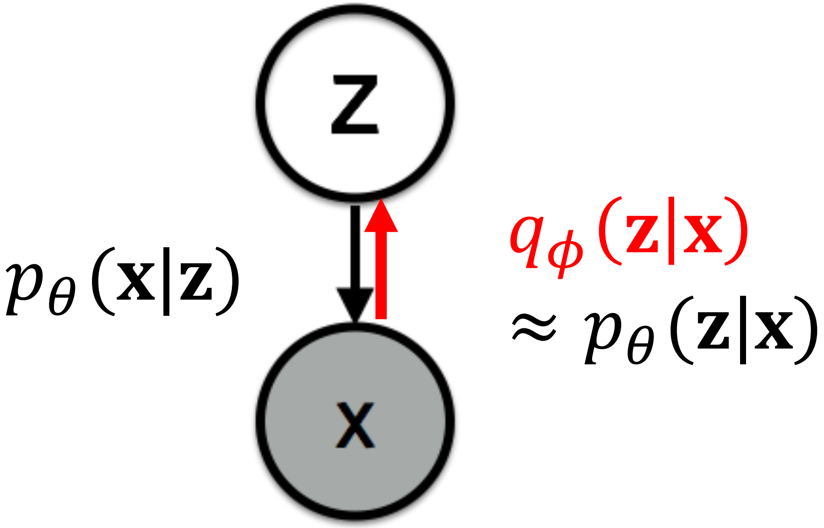
$$\log \prod_{\mathbf{x} \in \mathcal{D}} p_\theta(\mathbf{x}) = \sum_{\mathbf{x} \in \mathcal{D}} \log p_\theta(\mathbf{x})$$

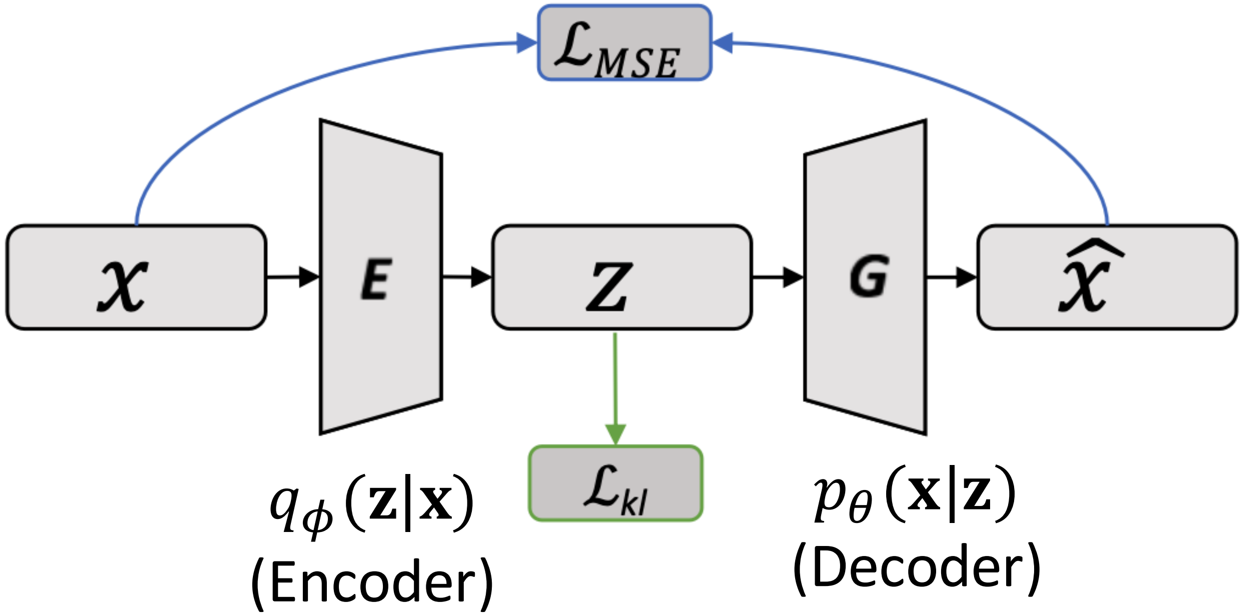
$$p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}|\mathbf{z}) p_\theta(\mathbf{z}) d\mathbf{z}$$

- Expensive to compute.

VAE: Variational Autoencoder

- Alternatively, we introduce a variational posterior $q_\phi(\mathbf{z}|\mathbf{x})$ to approximates the true posterior $p_\theta(\mathbf{z}|\mathbf{x})$?





VAE: Variational Autoencoder

- Use KL divergence to quantify the distance of these two posteriors:

$$
\begin{aligned}
& D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})) \\
& = \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}|\mathbf{x})} d\mathbf{z} \\
& = \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}) p_\theta(\mathbf{x})}{p_\theta(\mathbf{z}, \mathbf{x})} d\mathbf{z} \\
& = \int q_\phi(\mathbf{z}|\mathbf{x}) \big(\log p_\theta(\mathbf{x}) + \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}, \mathbf{x})} \big) d\mathbf{z} \\
& = \log p_\theta(\mathbf{x}) + \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}, \mathbf{x})} d\mathbf{z} \\
& = \log p_\theta(\mathbf{x}) + \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{x}|\mathbf{z}) p_\theta(\mathbf{z})} d\mathbf{z} \\
& = \log p_\theta(\mathbf{x}) + \mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} [\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z})} - \log p_\theta(\mathbf{x}|\mathbf{z})] \\
& = \log p_\theta(\mathbf{x}) + D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z})) - \mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z})
\end{aligned}
$$

VAE: Variational Autoencoder

- After expanding the equation:

$$D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x})) = \log p_\theta(\mathbf{x}) + D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})) - \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\log p_\theta(\mathbf{x}|\mathbf{z})$$

- Rearrange:

$$\boxed{\log p_\theta(\mathbf{x}) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x}))} = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\log p_\theta(\mathbf{x}|\mathbf{z}) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}))$$

Maximize during training

- Loss function:

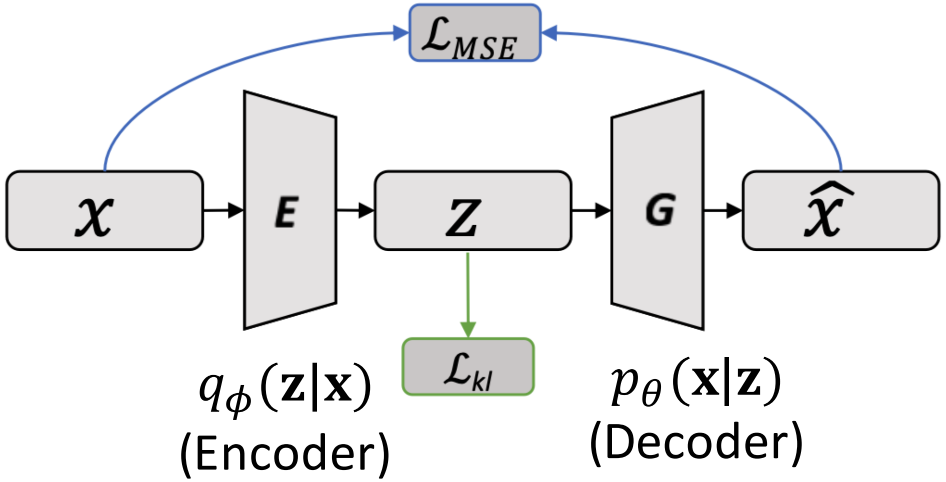
$$\begin{aligned} L_{\mathrm{VAE}}(\theta,\phi) &= -\log p_\theta(\mathbf{x}) + D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x})) \\ &= -\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\log p_\theta(\mathbf{x}|\mathbf{z}) + D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})) \end{aligned}$$

$$\theta^*,\phi^* = \arg\min_{\theta,\phi} L_{\mathrm{VAE}}$$

VAE: Variational Autoencoder

Loss function: Evidence Lower Bound (ELBO)

$$
\begin{aligned}
L_{\text{VAE}}(\theta, \phi) &= -\log p_\theta(\mathbf{x}) + D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})) \\
&= \underbrace{-\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z})}_{\text{Can be represented by MSE}} + \underbrace{D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}))}_{\text{regularisation}}
\end{aligned}
$$



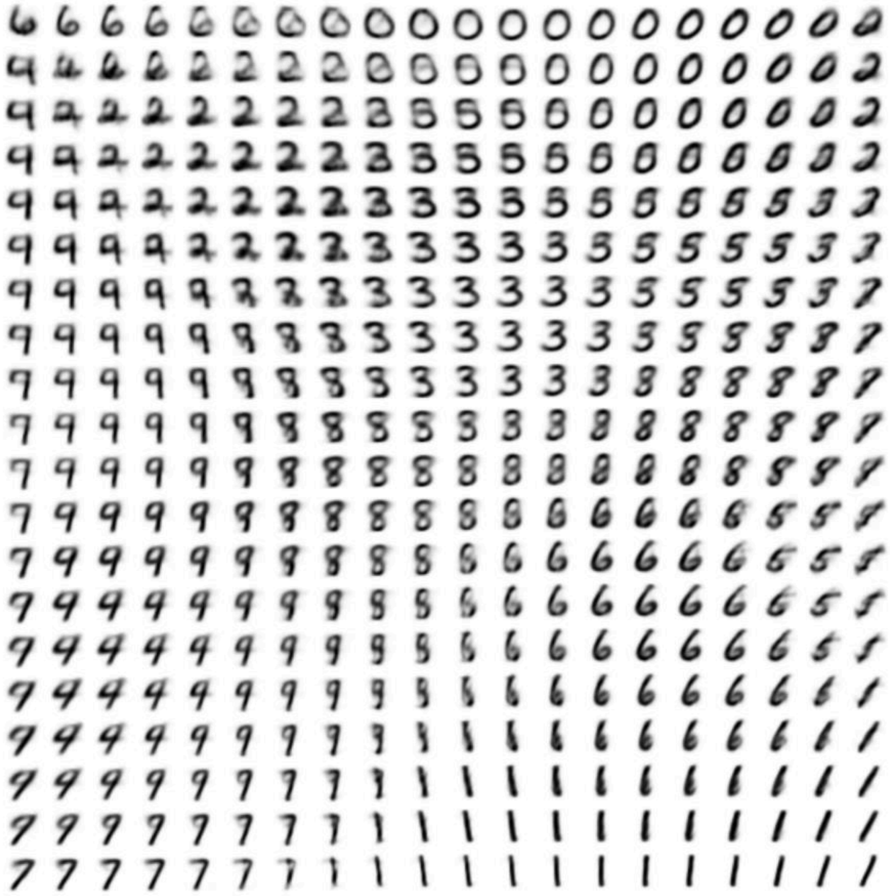
- Since $D_{\text{KL}}\big(q_\phi(\mathbf{z}|\mathbf{x})\big|p_\theta(\mathbf{z}|\mathbf{x})\big) \geq 0$, $\log p_\theta(x) \geq -L_{VAE}$.
- $-L_{VAE}$ is the lower bound of $\log p_\theta(x)$

VAE: Variational Autoencoder

Autoencoder VS. VAE

- AE: feature representation, $\mathbf{z}$ = encoder($\mathbf{x}$) is deterministic
- VAE : distribution representation, $p_\theta(\mathbf{z}|\mathbf{x})$ is a distribution

Results of VAE

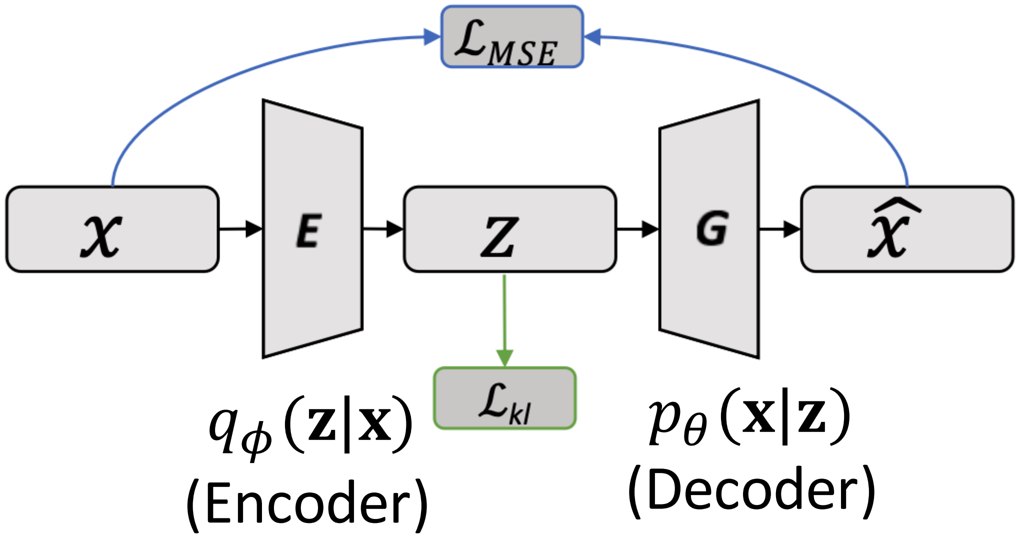


[Wojciech Mormul on Github](#) & Autoencoding beyond pixels using a learned similarity metric

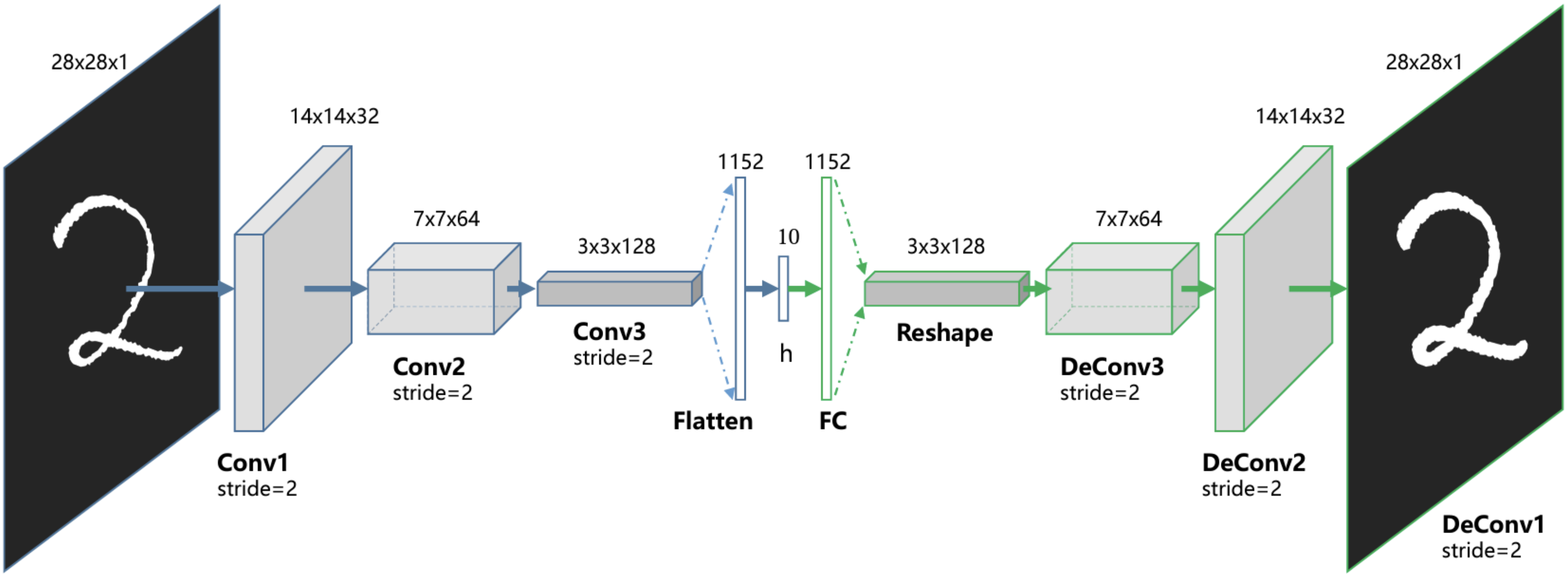
Convolutional VAE

Limitations of vanilla VAE

- The size of weight of fully connected layer = input size x output size
- If VAE uses fully connected layers only, will lead to curse of dimensionality when the input dimension is large (e.g., image).



Convolutional VAE

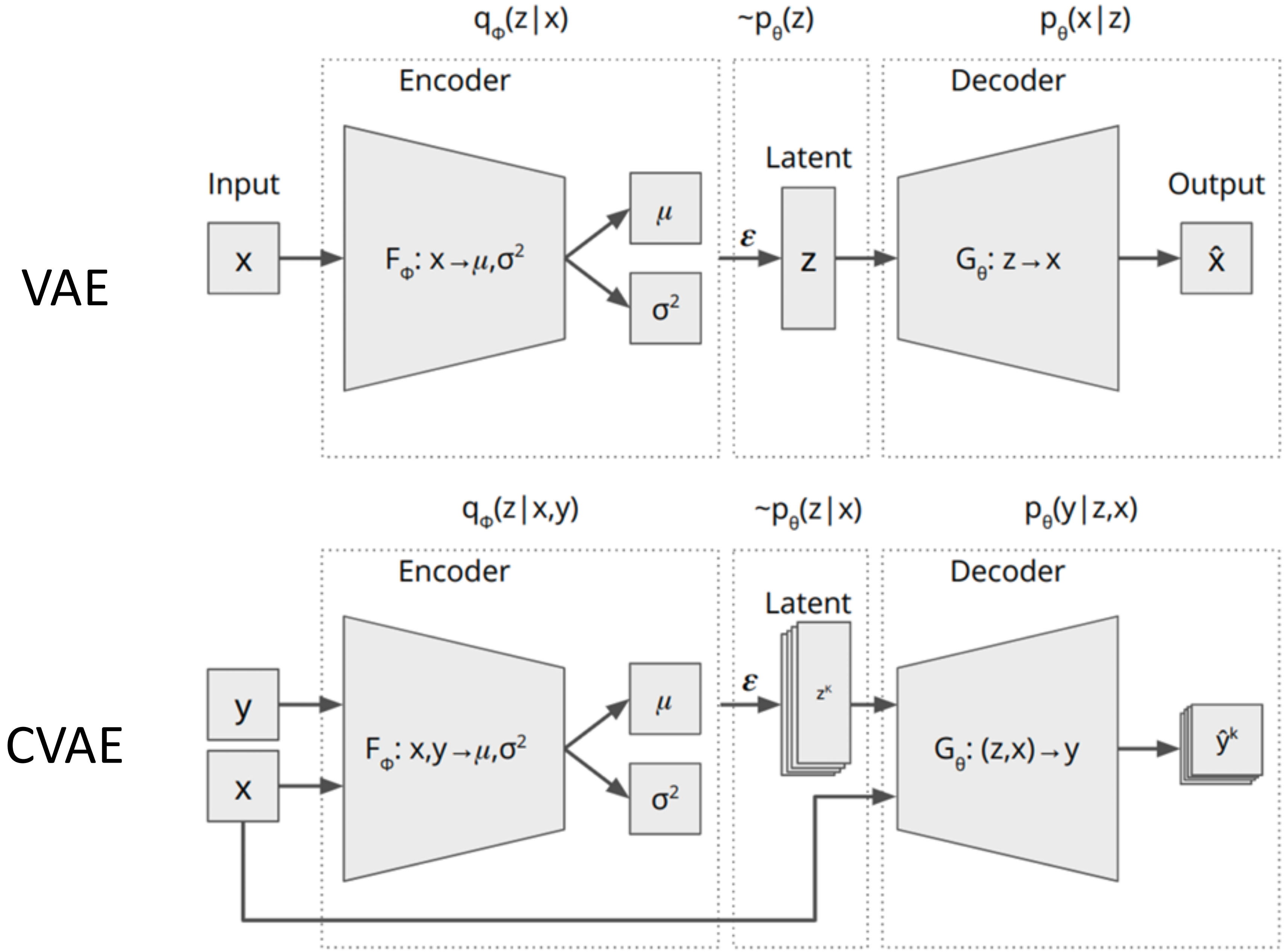


Deep Clustering with Convolutional Autoencoder. NIPS 2017.

Conditional VAE

What if we have labels? (e.g. digit labels or attributes) Or other inputs we wish to condition on ($\mathbf{y}$).

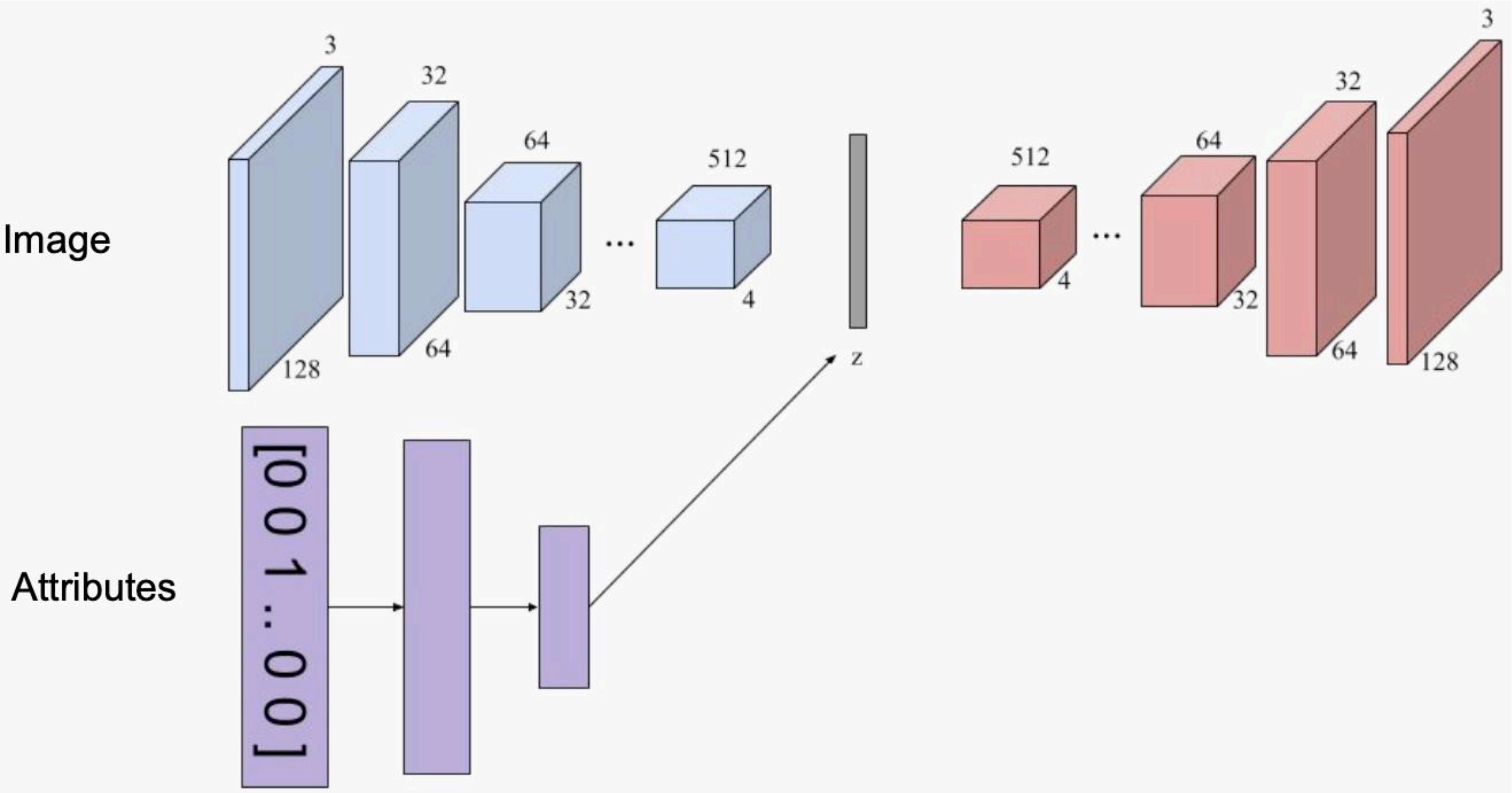
- None of the derivation changes.
- Replace all $p(\mathbf{x}|\mathbf{z})$ with $p(\mathbf{x}|\mathbf{z}, \mathbf{y})$.
- Replace all $q(\mathbf{z}|\mathbf{x})$ with $q(\mathbf{z}|\mathbf{x}, \mathbf{y})$.
- Go through the same KL divergence procedure, to get the same lower bound.



Learning structured output representation using deep conditional generative models.

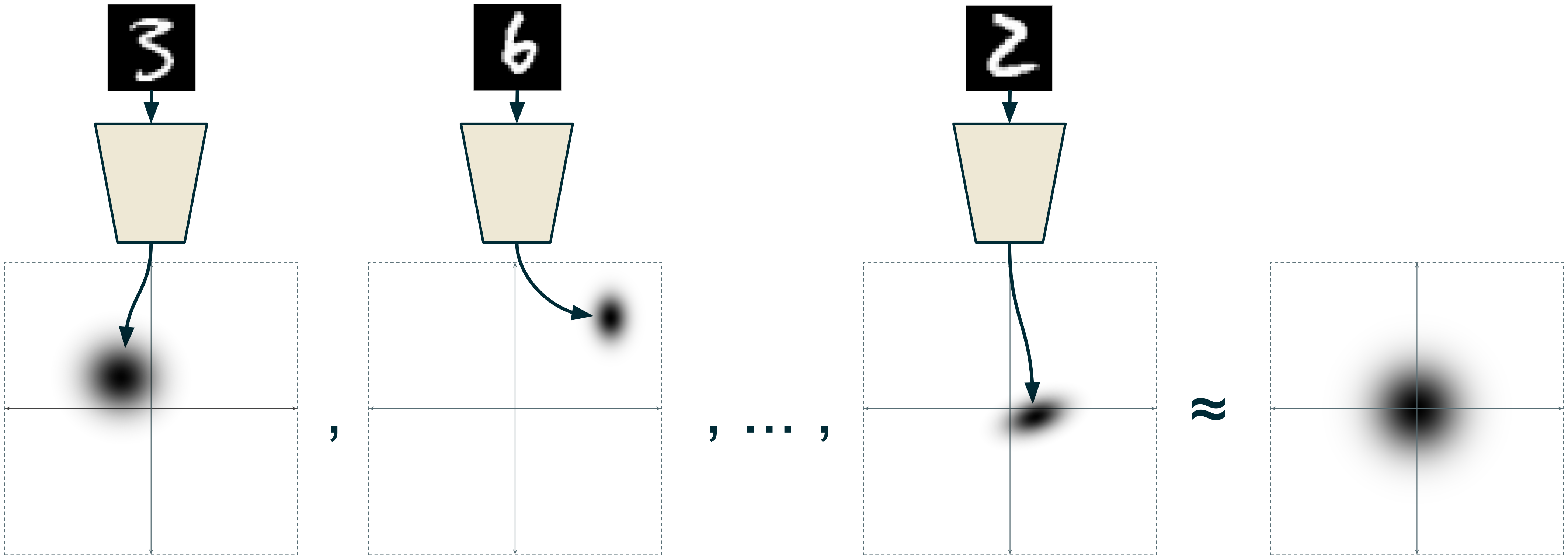
Conditional VAE

Common Architecture (convolutional)



Conditional VAE

- Train and inference without labelled data i.e., vanilla VAE

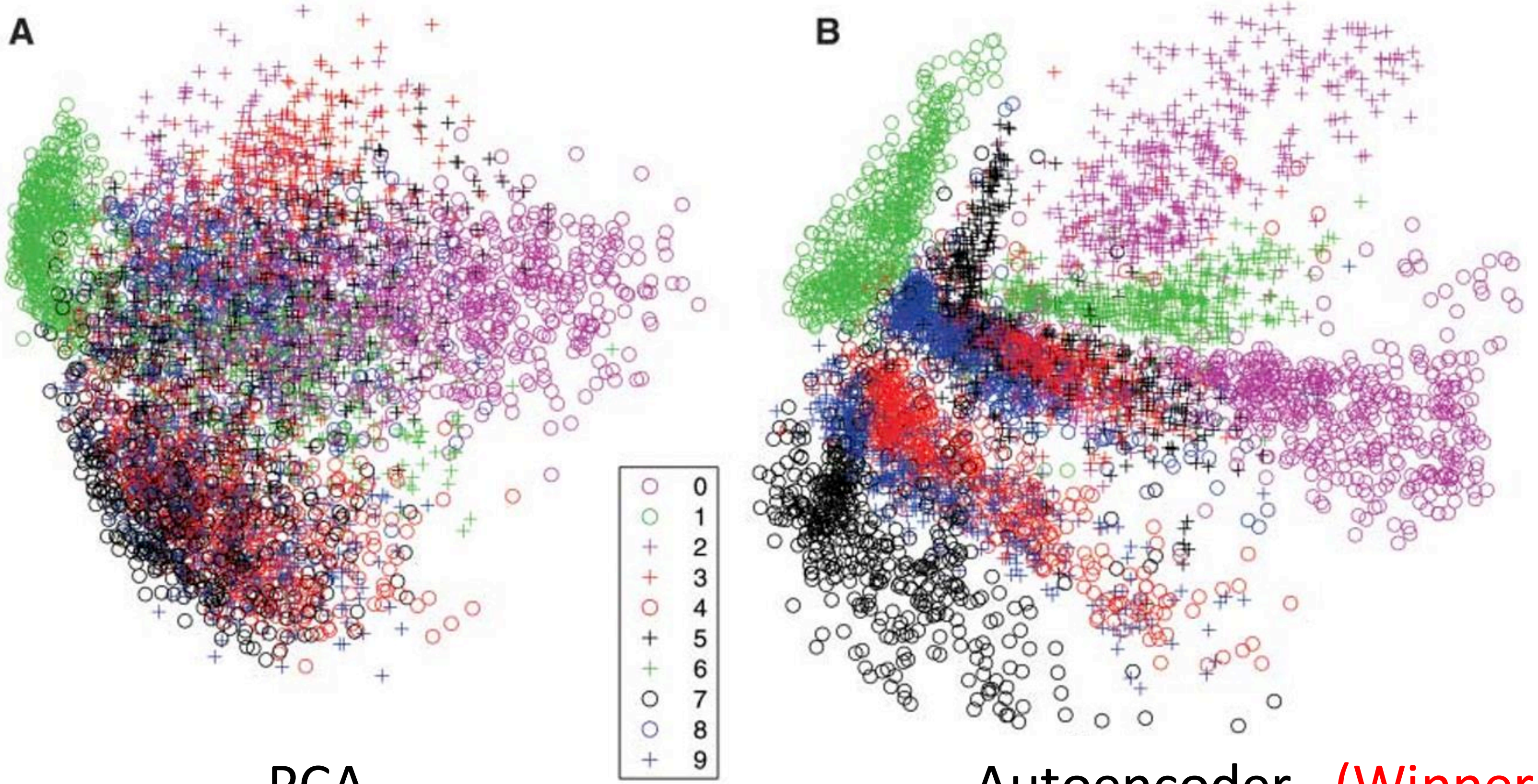


https://ijdykeman.github.io/ml/2016/12/21/cvae.html

Vanilla Autoencoder (previous slide)

Power of Latent Representation: t-SNE visualization on MNIST

Fig. 3. (**A**) The two-dimensional codes for 500 digits of each class produced by taking the first two principal components of all 60,000 training images. (**B**) The two-dimensional codes found by a 784-1000-500-250-2 autoencoder. For an alternative visualization, see (*8*).



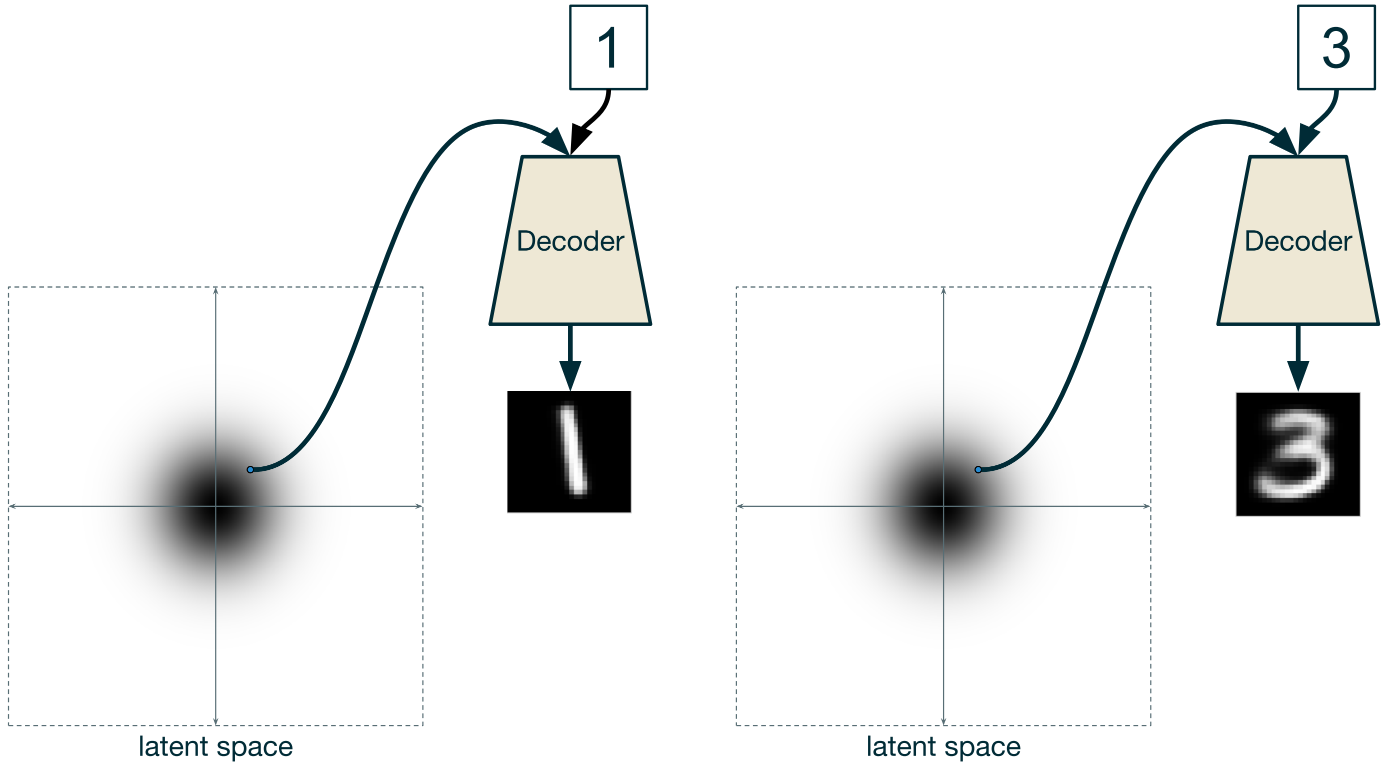
PCA

Autoencoder (Winner)

Hinton and Salakhutdinov , Reducing the Dimensionality of Data with Neural Networks, 2006

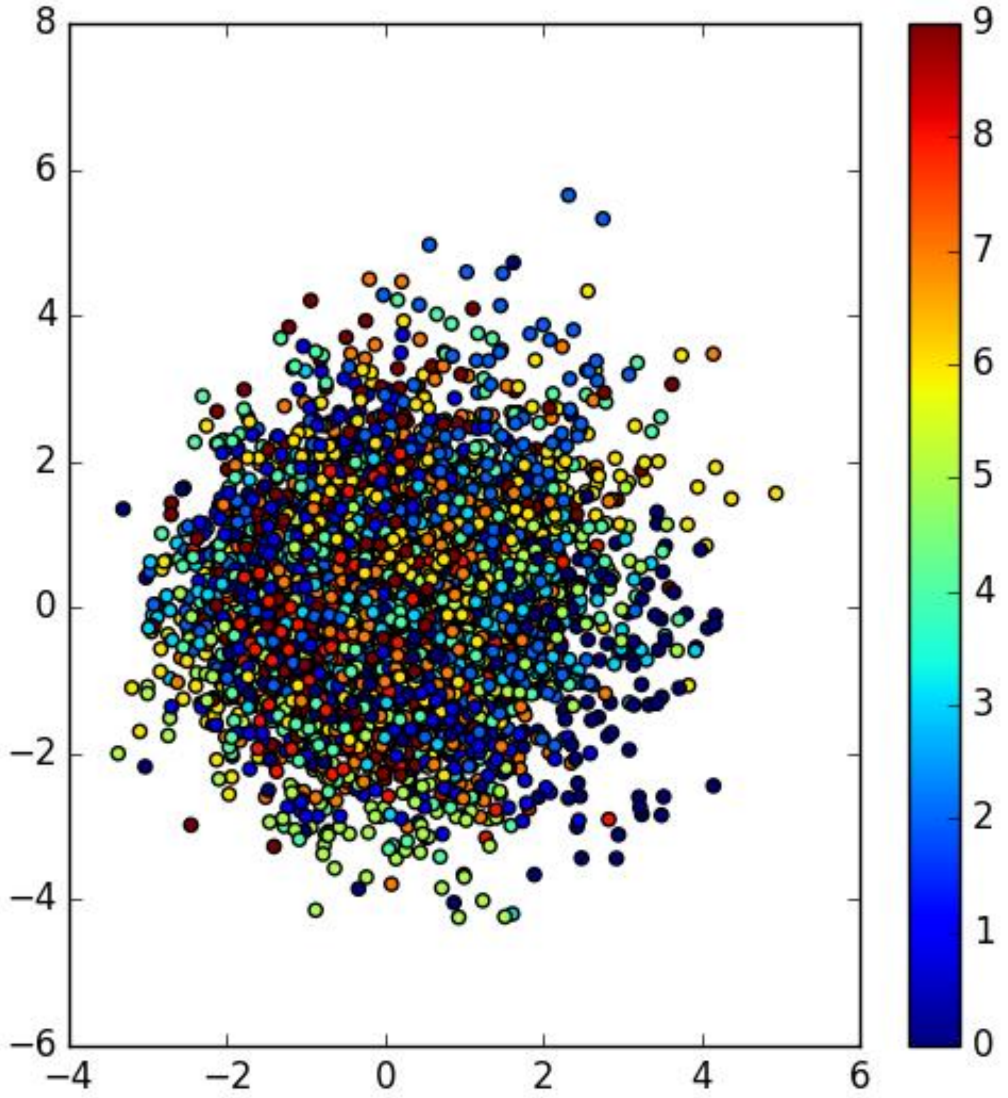
Conditional VAE

- Train and inference with labelled data



Conditional VAE on MNIST

- Generate MNIST data, conditioned to its label ($\mathbf{y}$):
- Visualize $q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{y})$



https://agustinus.kristia.de/techblog/2016/12/17/conditional-vae/

Conditional VAE on MNIST

- Reconstruct results

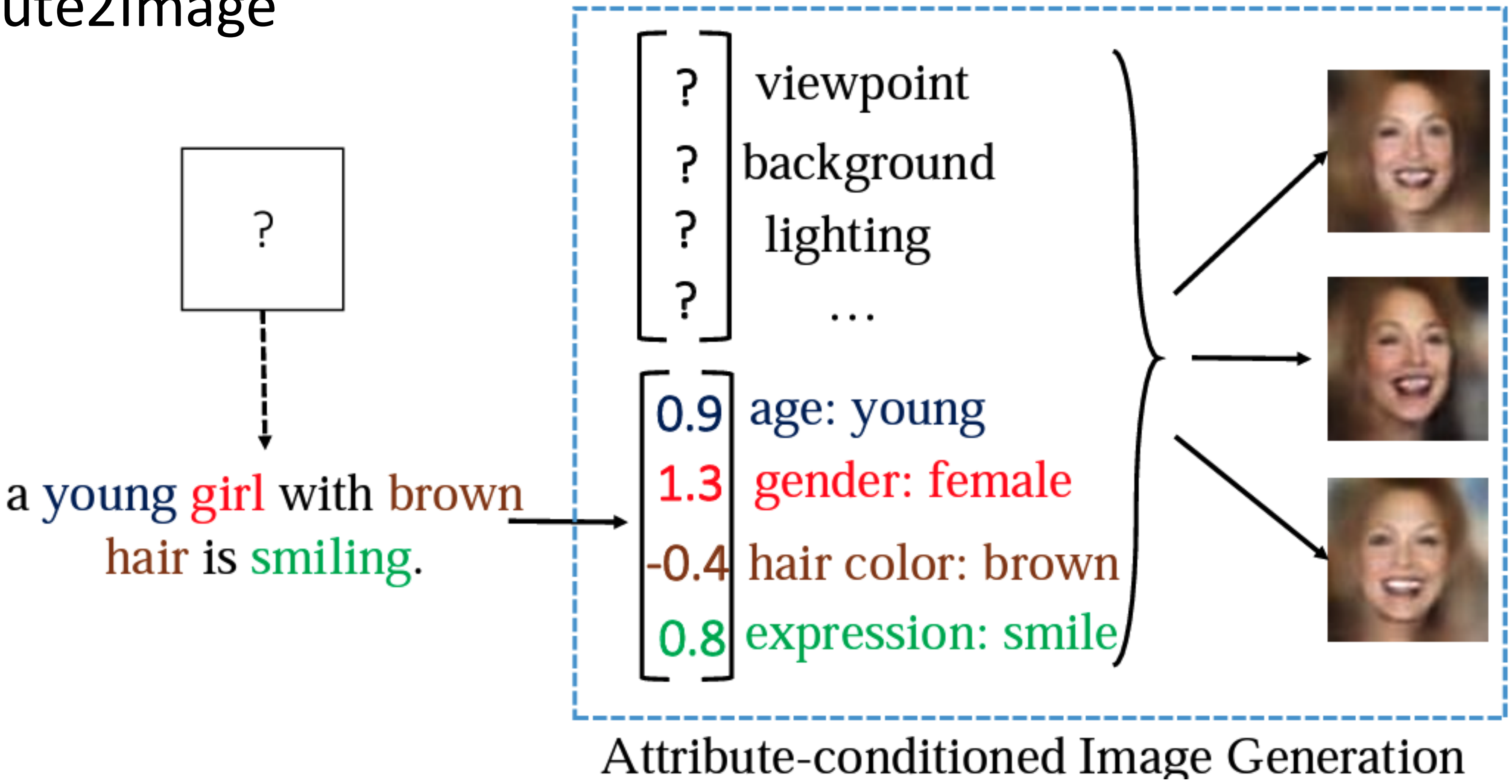
- Conditioned generation results

Conditional VAE Applications

- Attribute2Image
- Diverse Colorization

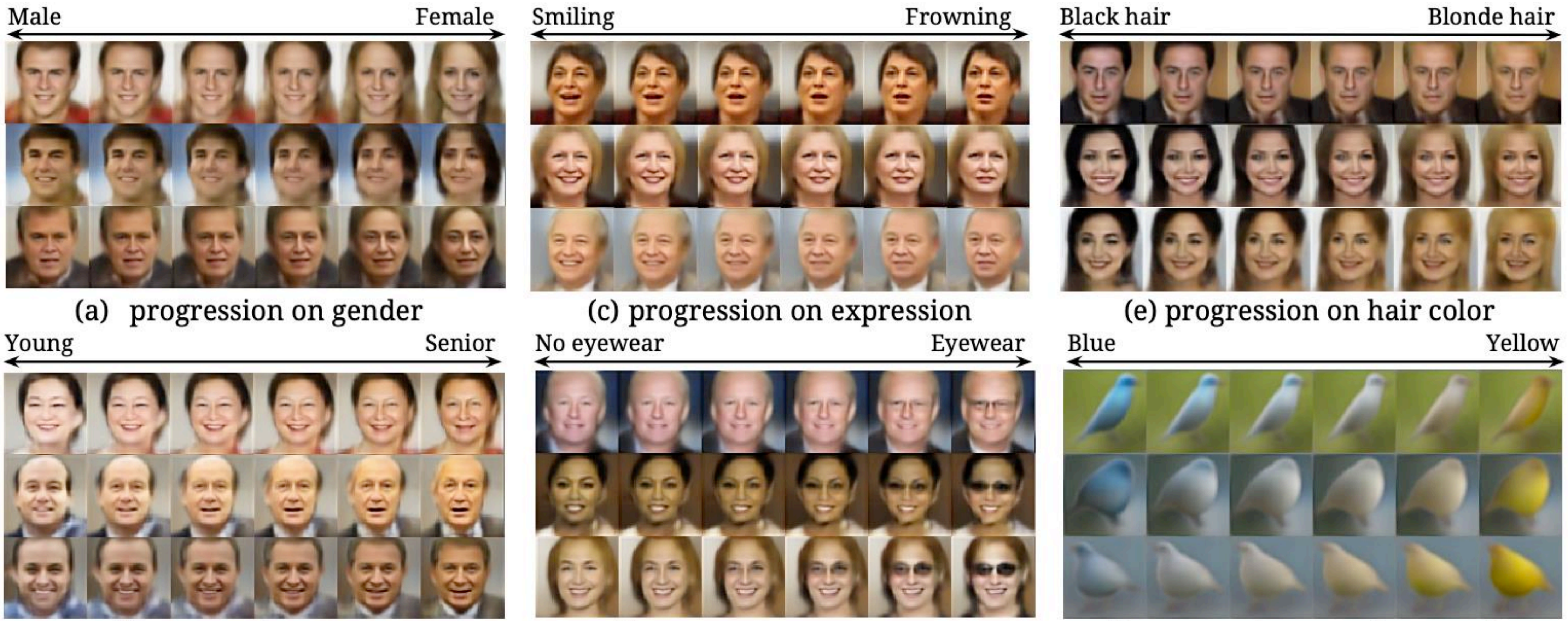
Conditional VAE Applications

Attribute2Image



Attribute2Image: Conditional Image Generation from Visual Attributes

Conditional VAE Applications



(a) progression on gender
(b) progression on age
(c) progression on expression
(d) progression on eyewear
(e) progression on hair color
(f) progression on primary color

$$p_\theta(x|y,z) \text{ with } z \sim \mathcal{N}(0,I) \text{ and } y = [y_\alpha, y_{rest}], \text{ where } y_\alpha = (1-\alpha)\cdot y_{min} + \alpha \cdot y_{max}$$

Attribute2Image: Conditional Image Generation from Visual Attributes

Conditional VAE Applications

- Image Colorization

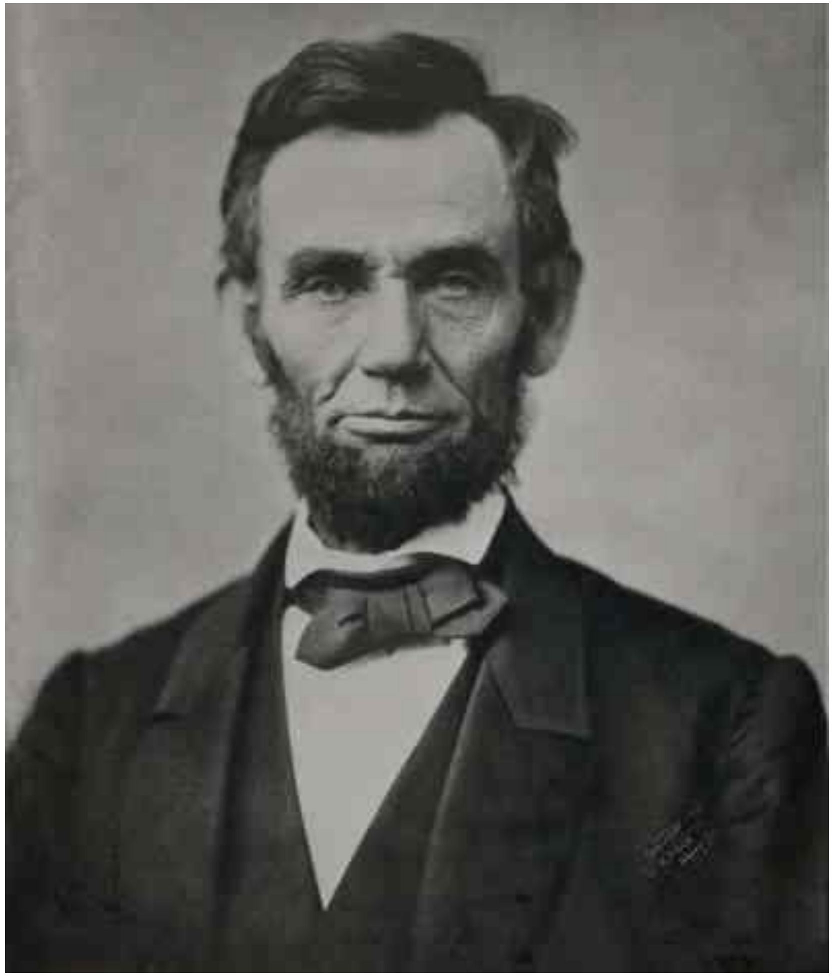
An ambiguous problem

Blue?
Red?
Yellow?

Conditional VAE Applications

- Image Colorization
- Goal: Learn a conditional model P(C|G)

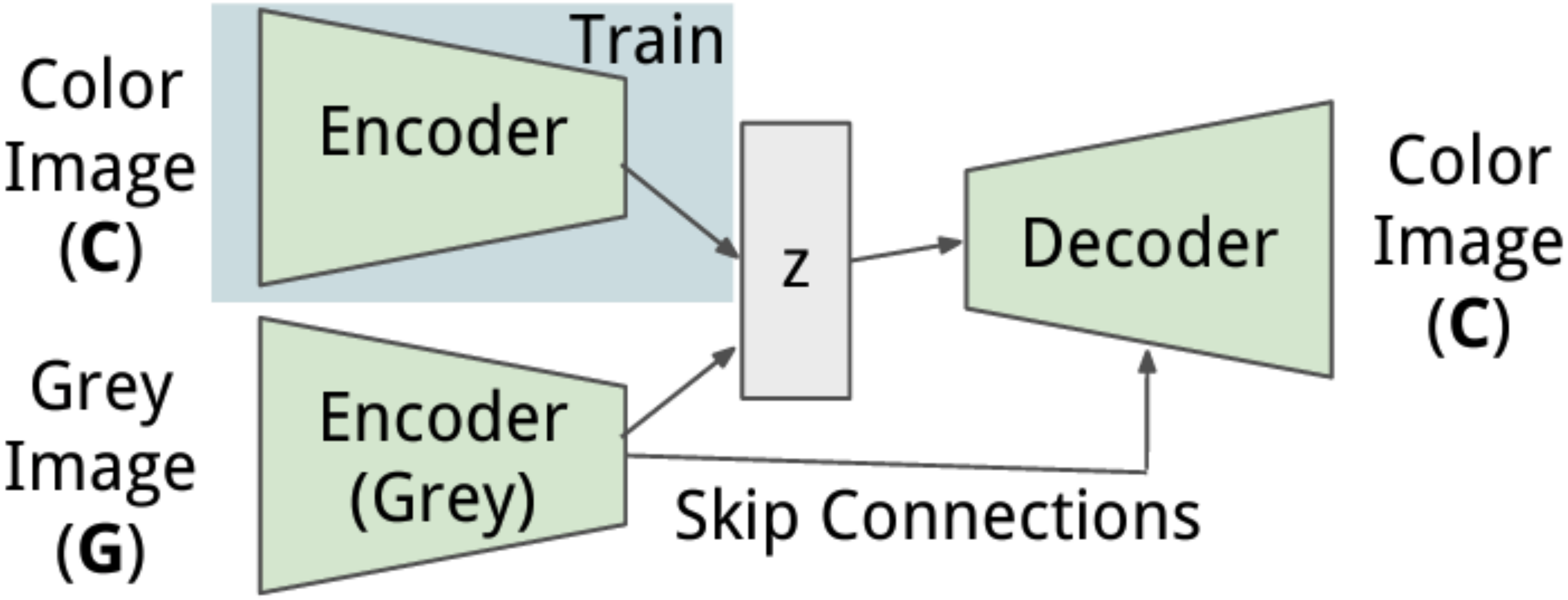
  (Color field C, given grey level image G)
- Next, draw samples from {C} ~ P(C|G) to obtain diverse colorization



Deshpande et al., Learning Diverse Image Colorization, CVPR 2017

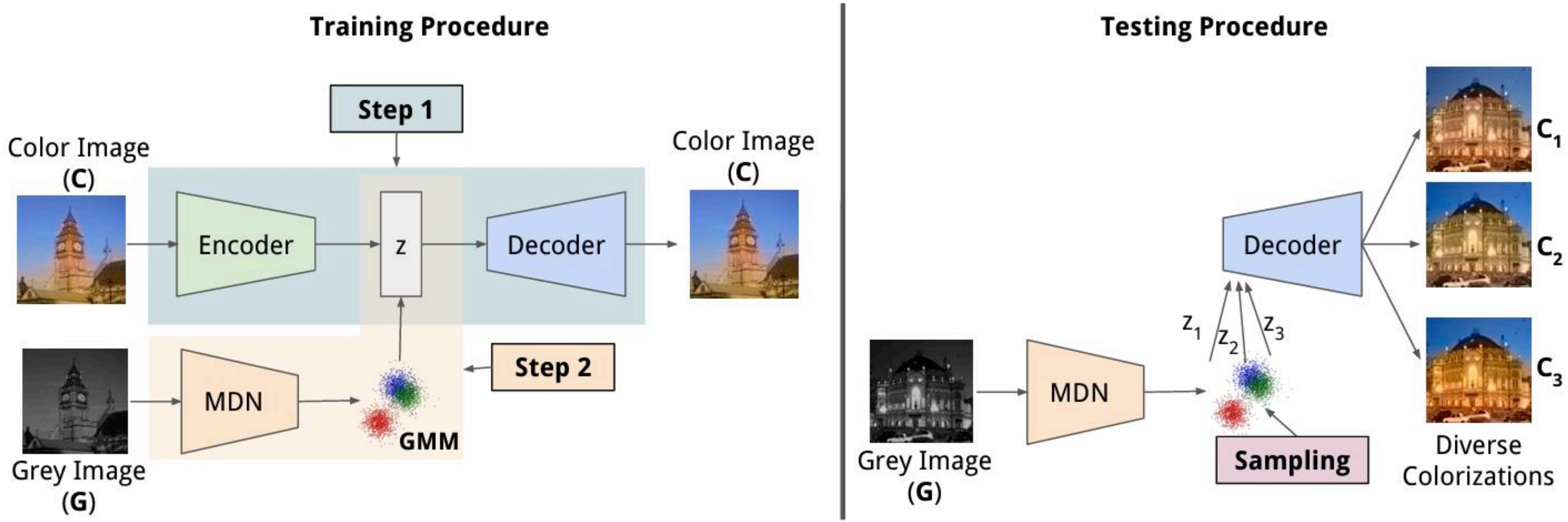
Conditional VAE Applications

- CVAE baseline



Deshpande et al., Learning Diverse Image Colorization, CVPR 2017

Conditional VAE Applications



Deshpande et al., Learning Diverse Image Colorization, CVPR 2017

Conditional VAE Applications

Step 1: Learn a low dimensional z for color.

- Standard VAE: Overly smooth, as training using L2 loss directly on the color space.
- Authors introduced several new loss functions to solve this problem.
  1. Weighted L2 on the color space to encourage ``color'' diversity. Weighting the very common color smaller.
  2. Top-k principal components, Pk, of the color space. Minimize the L2 of the projection.
  3. Encourage color fields with the same gradient as ground truth.

$$\mathcal{L}_{dec} = \boxed{\mathcal{L}_{hist}} + \lambda_{mah} \boxed{\mathcal{L}_{mah}} + \lambda_{grad} \boxed{\mathcal{L}_{grad}}$$

Deshpande et al., Learning Diverse Image Colorization, CVPR 2017

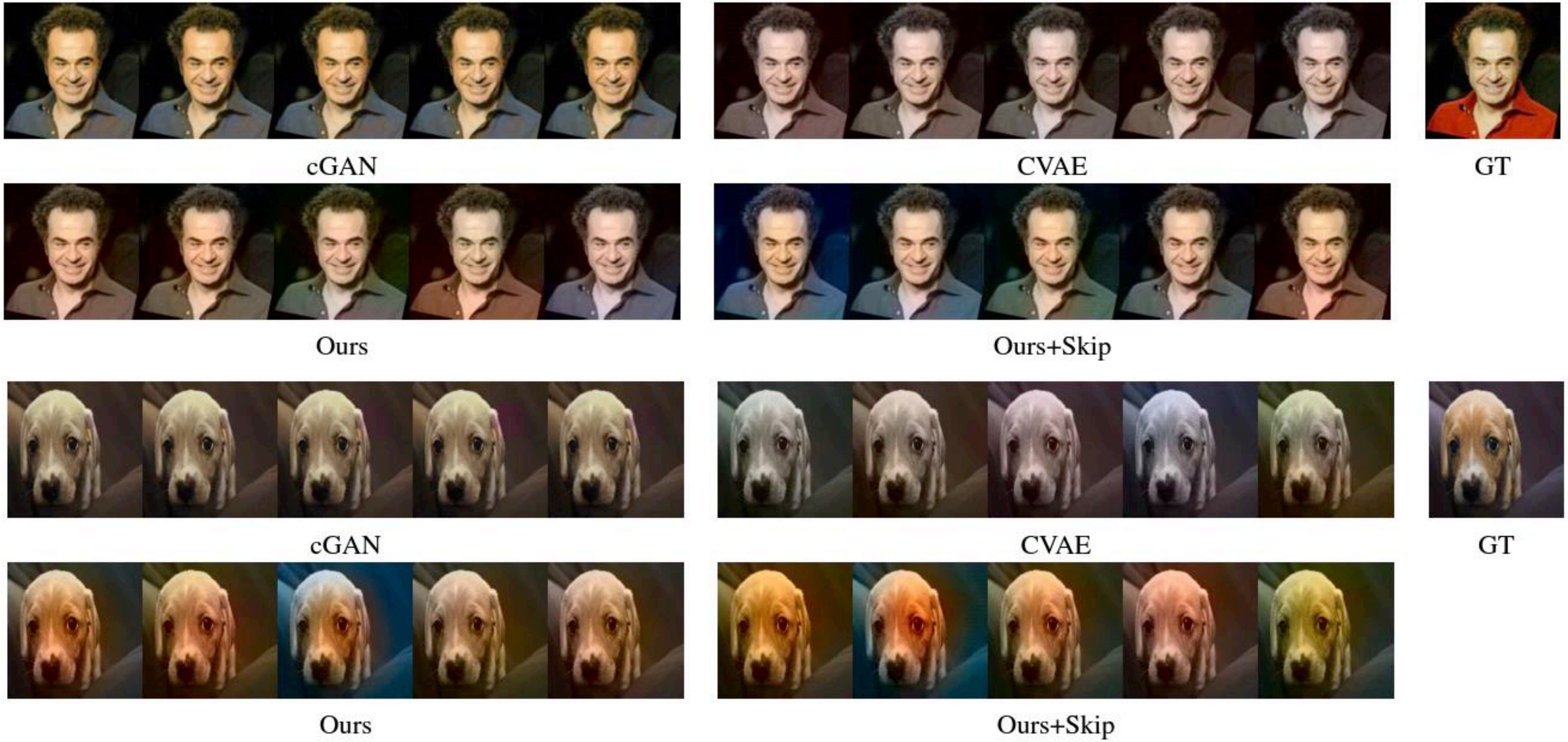
Conditional VAE Applications

- Step 2: Conditional Model: Grey-level to Embedding

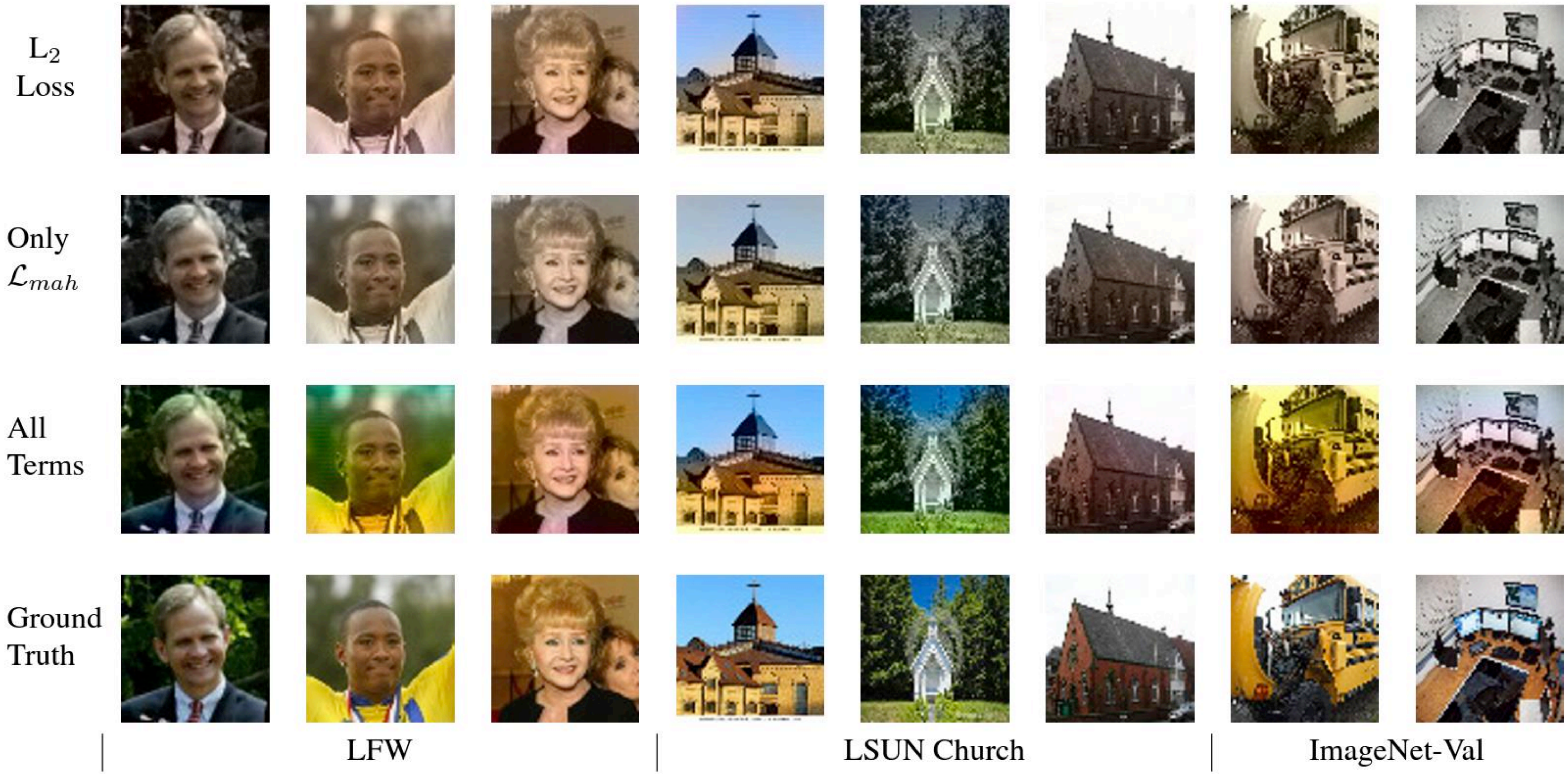
$$\mathcal{L}_{mdn} = -\log P(\mathbf{z}|\mathbf{G}) = -\log \sum_{i=1}^{M} \pi_i(\mathbf{G},\phi)\mathcal{N}(\mathbf{z}|\mu_i(\mathbf{G},\phi),\sigma)$$

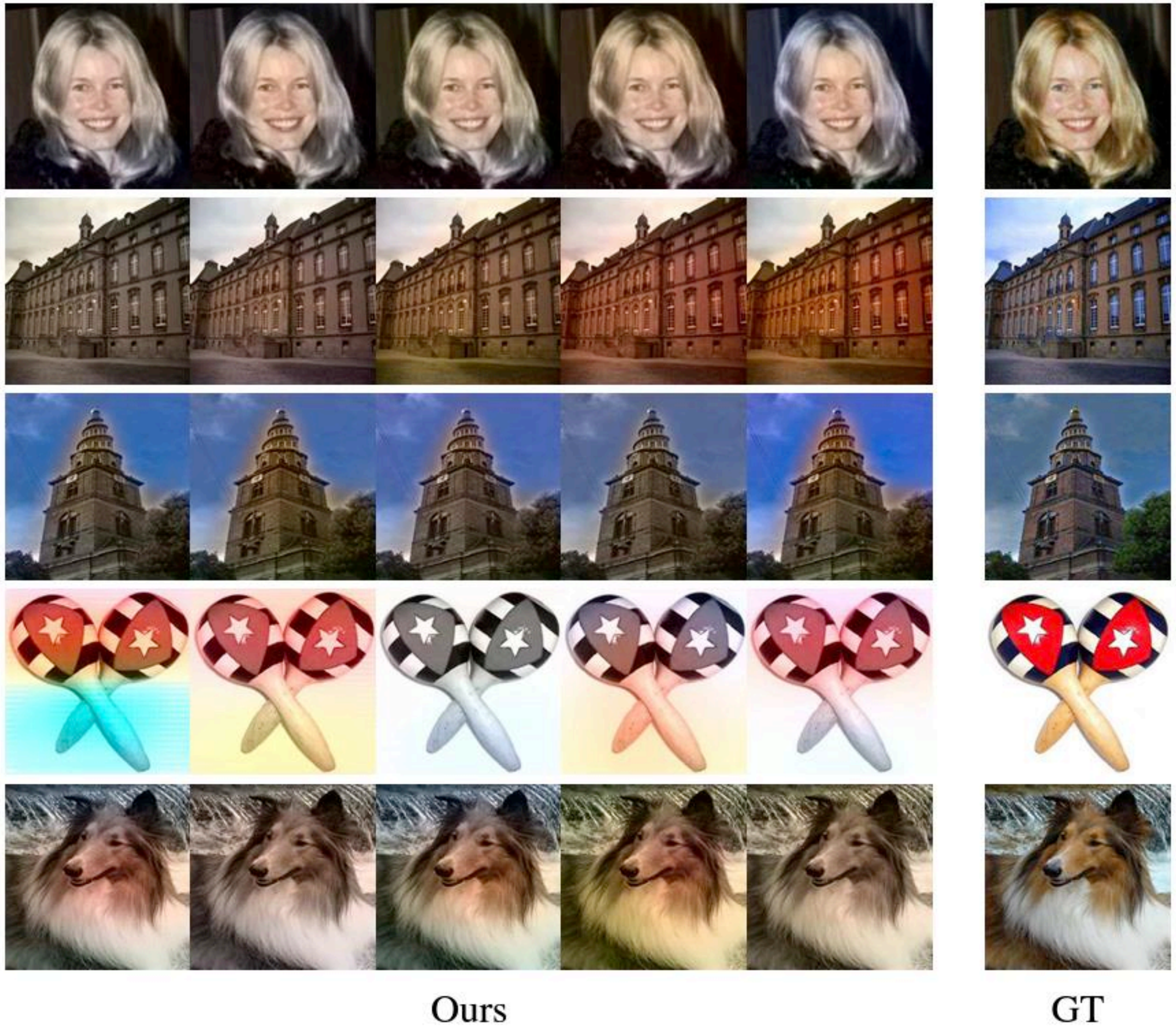
- Learn a multimodal distribution
- At test time sample at each mode to generate diversity.
- Similar to CVAE, but this has more “explicit” modeling of the P(z|G).

Conditional VAE Applications



Deshpande et al., Learning Diverse Image Colorization, CVPR 2017


L_2 Loss
Only $\mathcal{L}_{mah}$
All Terms
Ground Truth
LFW
LSUN Church
ImageNet-Val


Ours
GT

Thank You

- Questions?

- Email: yu.yin@case.edu