Obscuration of image content via latent space manipulation

Valentin NOYÉ, William PUECH, Pauline PUTEAUX, Norman HUTTE

LIRMM, Univ Montpellier, CNRS

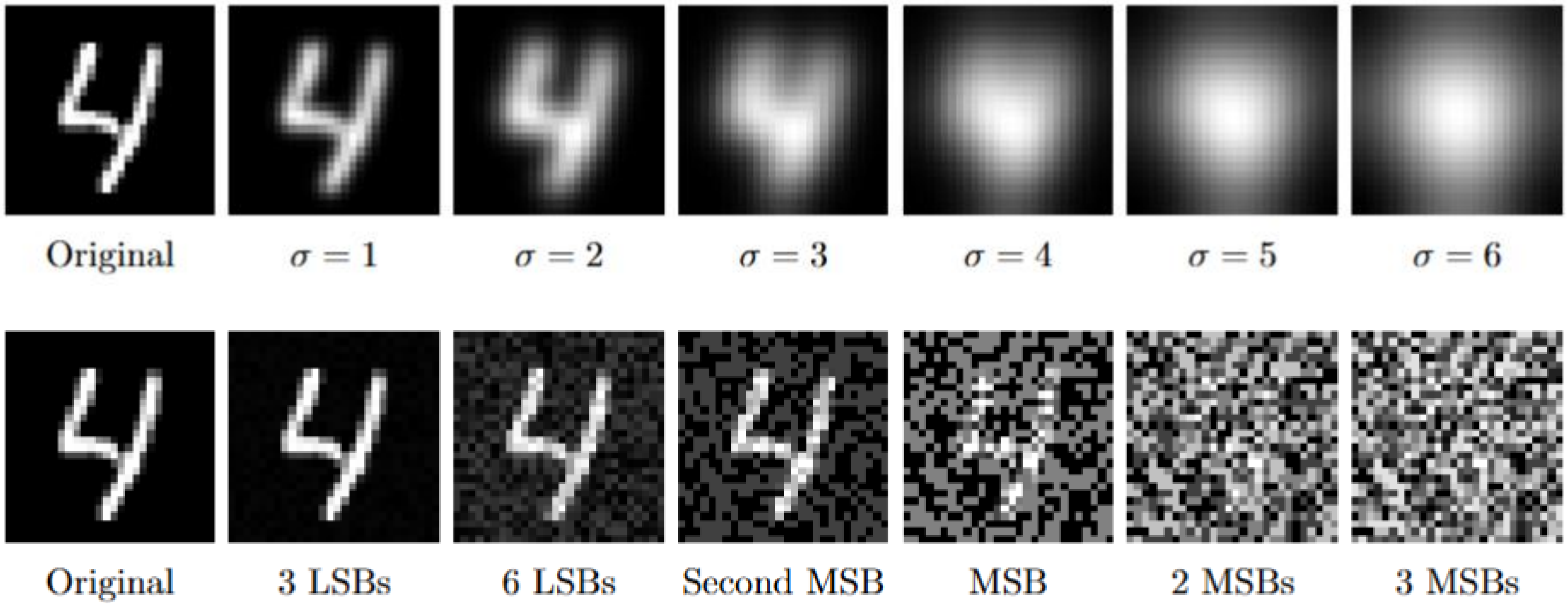
Context

- Importance of information in visual media
- Data and **privacy protection**
- Good **perceptual quality** and **invisible**
- **Reversible obscuration** using a key

Overview of obscuration methods and limits

Blurring

Selective encryption (AES)



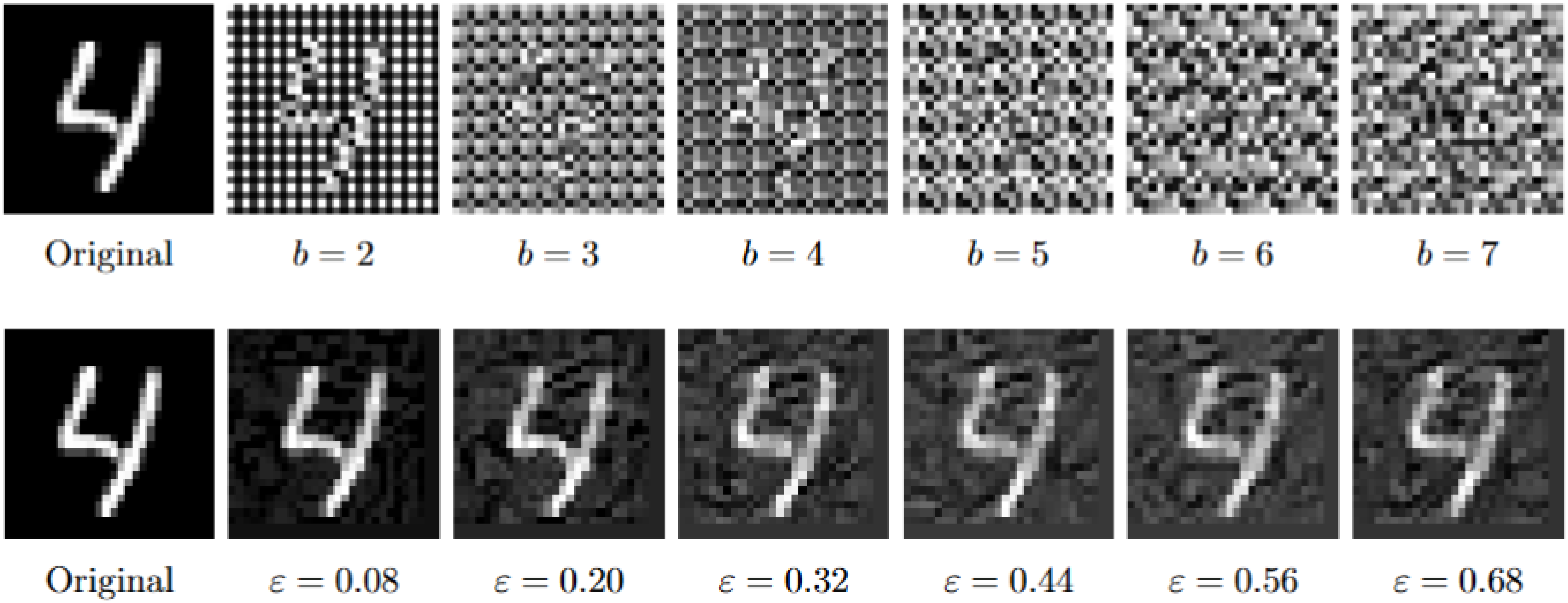
📖 **Steven Hill, Zhimin Zhou, Lawrence Saul, Hovav Shacham**
On the (In)effectiveness of Mosaicing and Blurring as Tools for Document Redaction
Proceedings on Privacy Enhancing Technologies, 2016

📖 **National Institute of Standards and Technology (NIST)**
Advanced Encryption Standard (AES)
FIPS Publication 197, 2001

Overview of obscuration methods and limits

Bit flipping

Poisoning (PGD)



📖 **MaungMaung AprilPyone, Hitoshi Kiya**
Block-Wise Image Transformation With Secret Key for Adversarially Robust Defense
IEEE Transactions on Information Forensics and Security, 2021

📖 **Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, Adrian Vladu**
Towards Deep Learning Models Resistant to Adversarial Attacks
International Conference on Learning Representations (ICLR), 2015

Solution

Obscuration

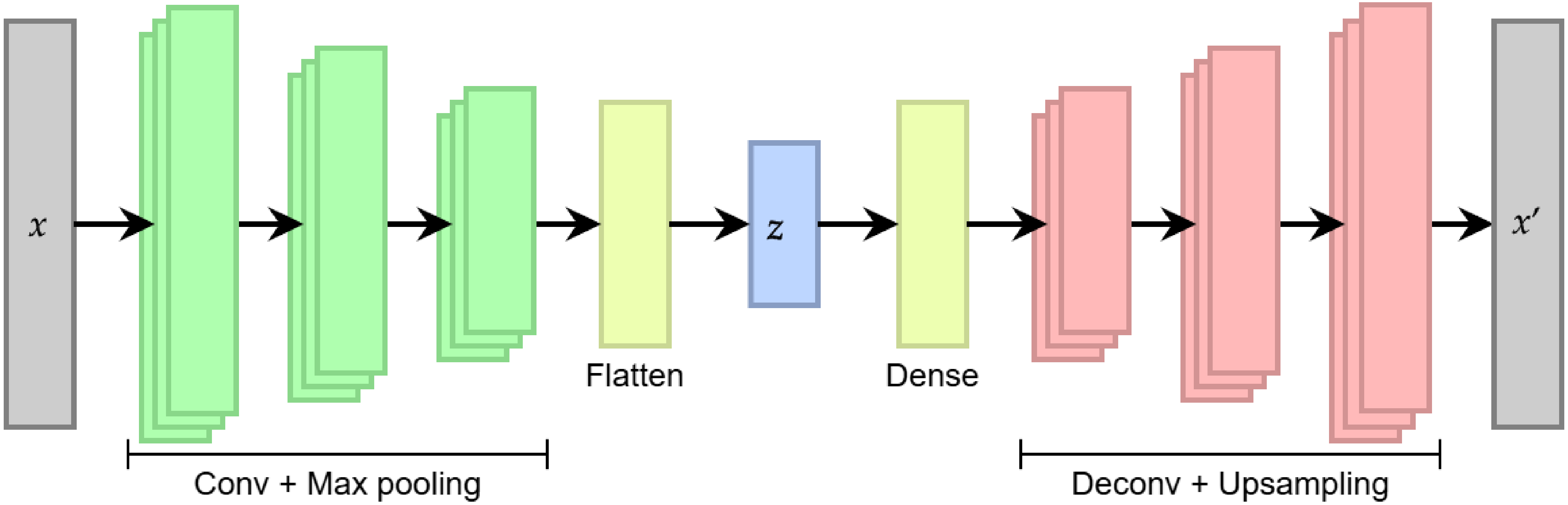
Reconstruction

Summary

- Introduction
- Overview of autoencoders and variants
- Our proposed obscuration methods
- Experimental results
- Attack of our methods
- Conclusion and prospects

Summary

Architecture of an autoencoder



➔ Binary cross-entropy loss: $$\mathcal{L}_{\text{recon}} = -\sum_{i=1}^{|\mathcal{X}|} x_i \ln x_i' \; + \; (1 - x_i) \ln(1 - x_i')$$

Variational Autoencoder's (VAE) formulation of Bayesian inference

- **Evidence**

  $p(x)$

- ***A posteriori***

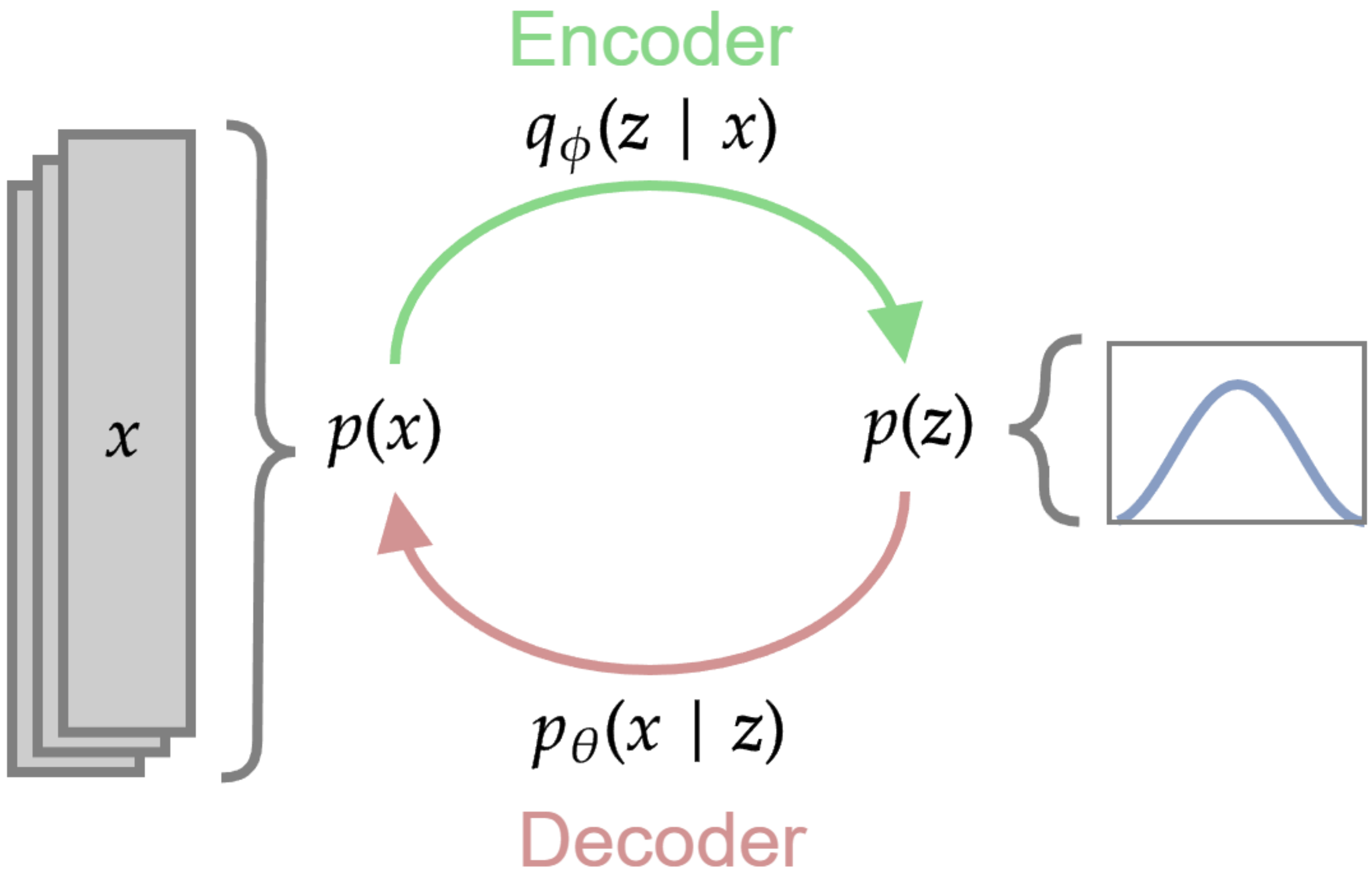
  $p_\varphi(\mathbf{z} \mid x) \approx q_\varphi(\mathbf{z} \mid x)$

- ***A priori***

  $p(\mathbf{z})$

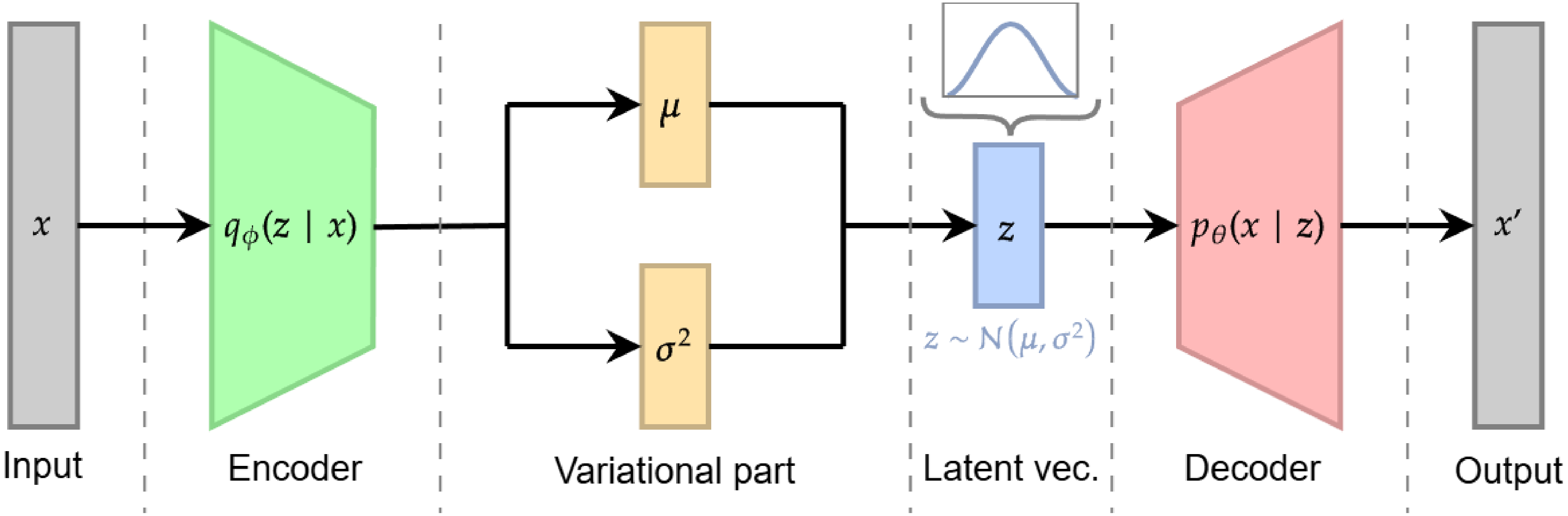
- **Likelihood**

  $p_\vartheta(x \mid \mathbf{z})$



📖 **Diederik P Kingma, Max Welling**
Auto-Encoding Variational Bayes
2nd International Conference on Learning Representations (ICLR), 2014

Architecture of a variational autoencoder (VAE)

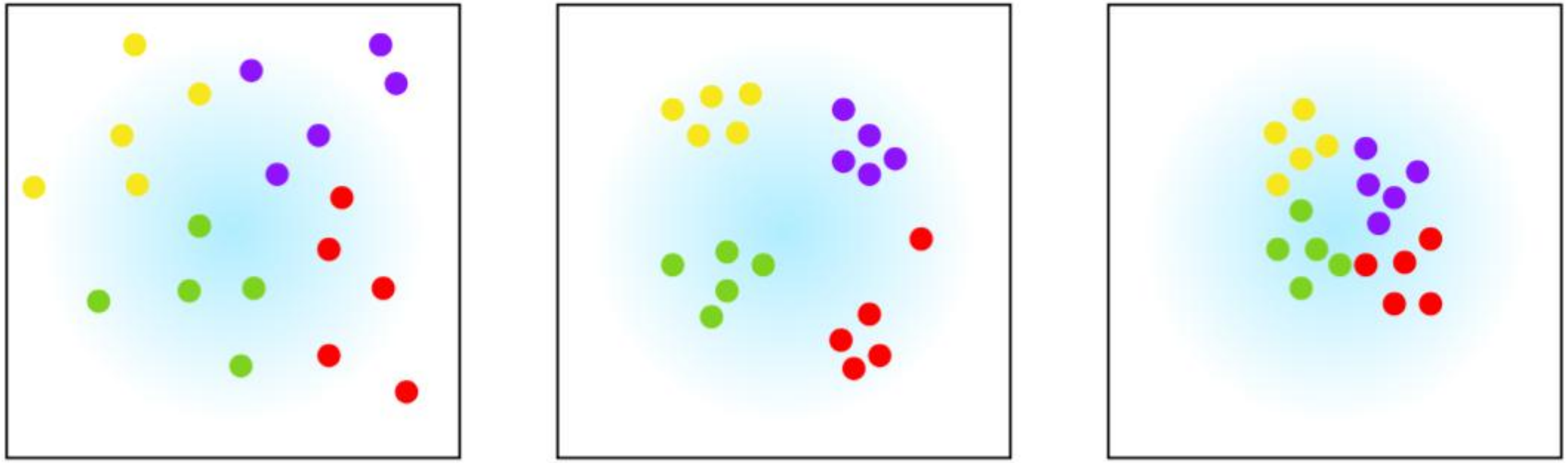


→ VAE loss: $\mathcal{L} = \underbrace{-\mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x \mid z)\right]}_{\mathcal{L}_{\text{recon}}} + \underbrace{D_{\text{KL}}\left(q_\phi(z \mid x) \,\|\, p(z)\right)}_{\mathcal{L}_{\text{KL}}}$

📖 **Diederik P Kingma, Max Welling**
Auto-Encoding Variational Bayes
2nd International Conference on Learning Representations (ICLR), 2014

Weighting the regularization of a VAE (β-VAE)

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

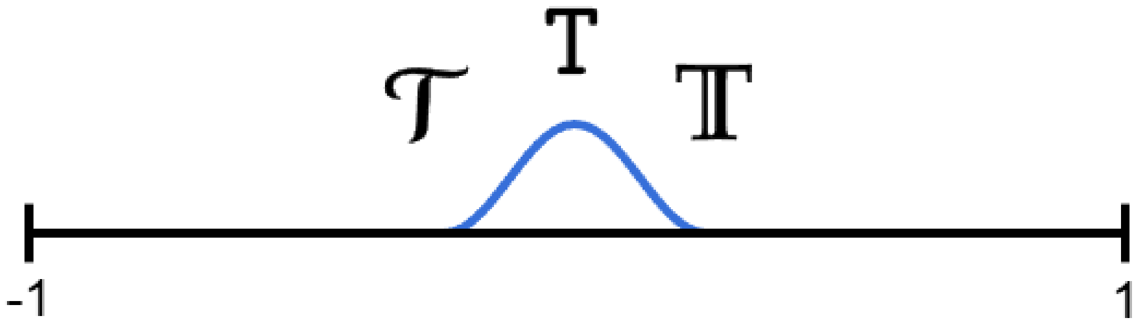


$\beta < 1$ (Standard autoencoder) $\quad$ $\beta = 1$ (VAE) $\quad$ $\beta > 1$ (β-VAE)

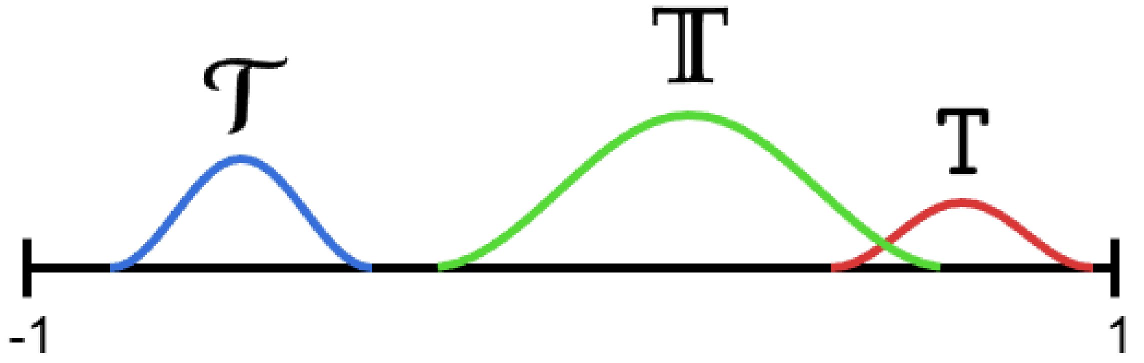
Christopher P. Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, Alexander Lerchner
Understanding disentangling in β-VAE
31st Conference on Neural Information Processing Systems (NIPS), 2017

On the modality of data distributions

- ➔ Information is **not always unimodal**.
- ➔ The data cannot be forced into a single gaussian when classes exhibit **more complex distributions**.

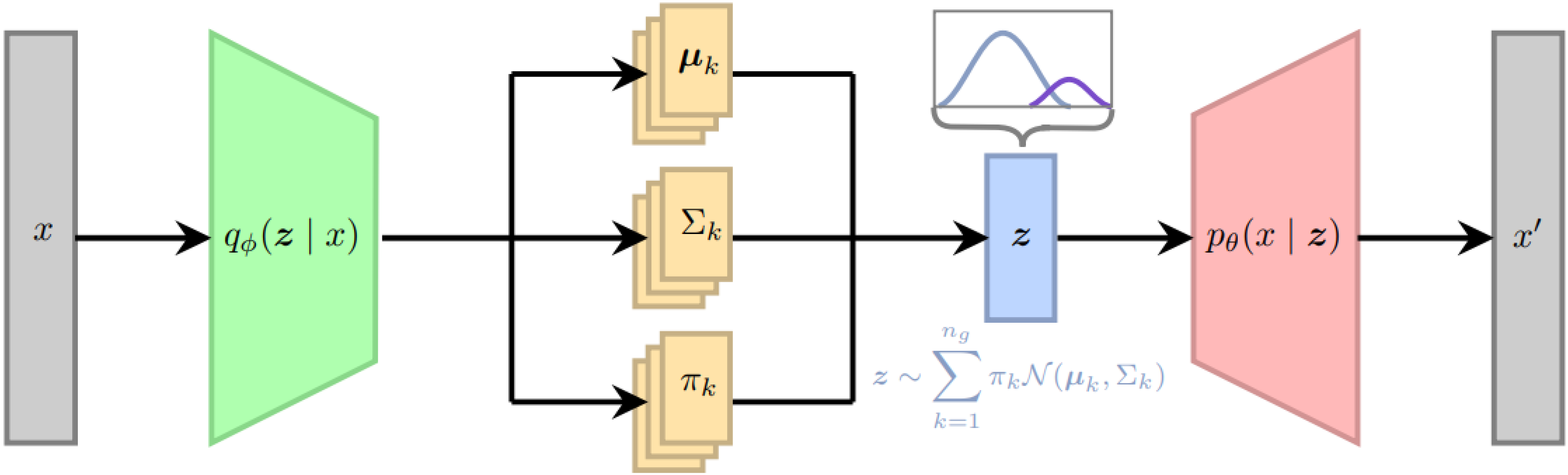


Gaussian *a priori*



Gaussian mixture *a priori*

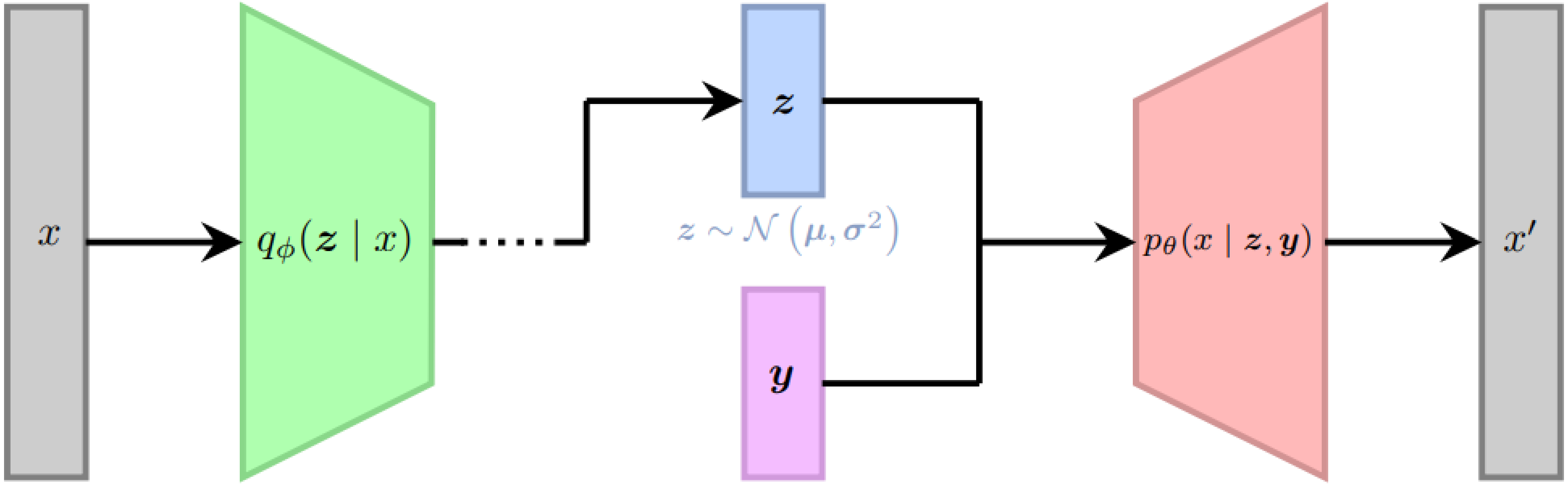
Gaussian Mixture VAE (GMVAE)



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

📖 **Nat Dilokthanakul, Pedro A.M. Mediano, Marta Garnelo, Matthew C.H. Lee, Hugh Salimbeni, Kai Arulkumaran, Murray Shanahan**
Deep Unsupervised Clustering with Gaussian Mixture Variational Autoencoders
31st Conference on Neural Information Processing Systems (NIPS), 2017

Conditional VAE (CVAE)



➔ **y** is **one-hot-encoded**. Example for 10 classes : C_3 = [0,0,0,1,0,0,0,0,0,0]

📖 **N. Siddharth, Brooks Paige, Jan-Willem van de Meent, Alban Desmaison, Noah D. Goodman, Pushmeet Kohli, Frank Wood, Philip H. S. Torr. L**
Learning Disentangled Representations With Semi-Supervised Deep Generative Models
31st Conference on Neural Information Processing Systems (NIPS), 2017

Summary

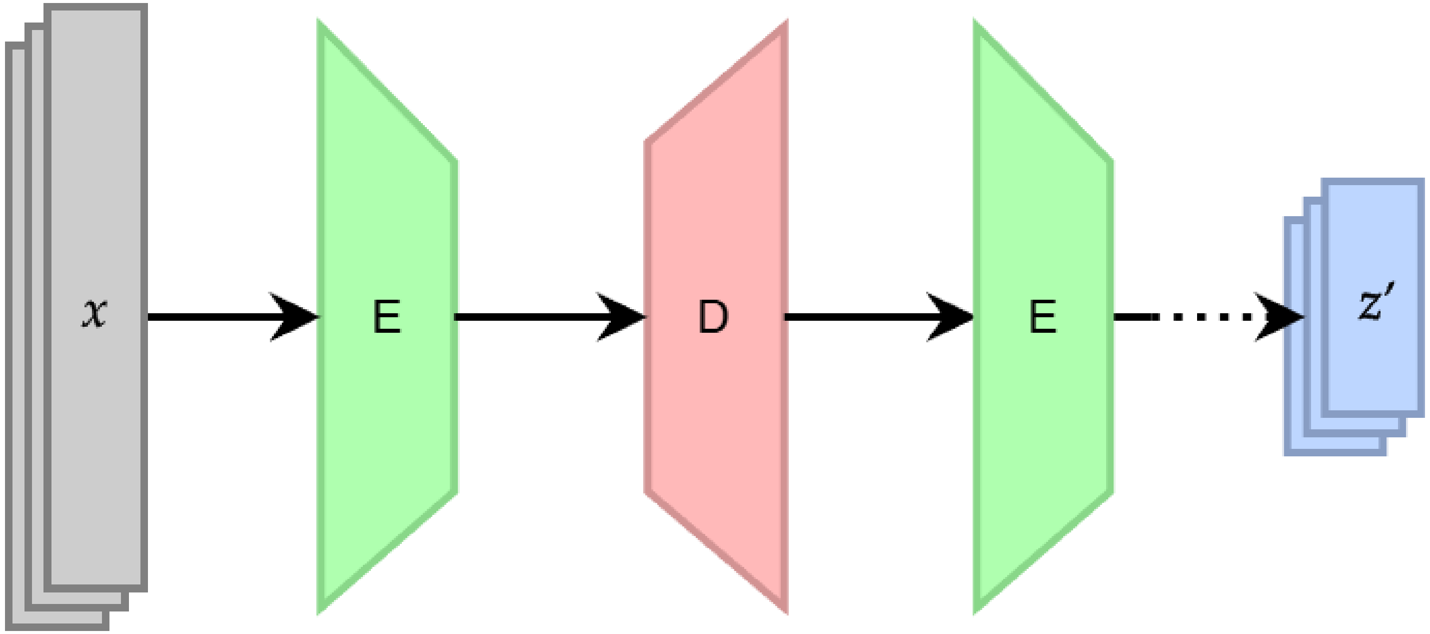
Introduction to obscuration methods

4 obscuration methods:

1. Method by **translation**: β-VAE or GMVAE
2. Method by **perturbed translation**: β-VAE or GMVAE
3. Method by **multivariate transformation**: β-VAE or GMVAE
4. Method by **conditioning**: CVAE

Autoencoder round-trip

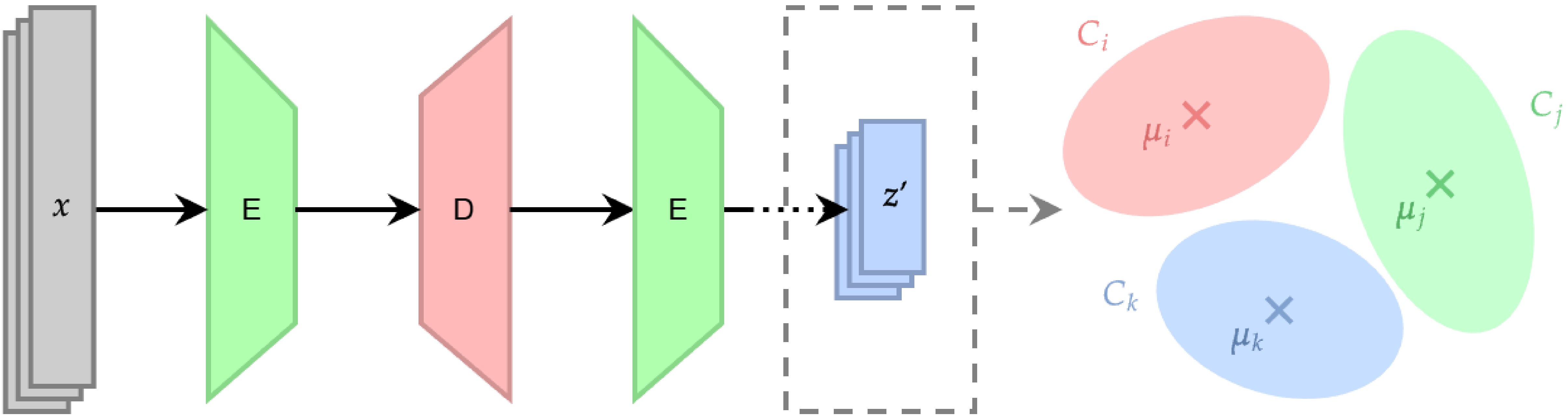
- → **Encode and decode** to align with the model's likelihood distribution
- → Then, **reencode** to obtain the latent vector **z′** to work on



How to obtain the distributions parameters?

→ **For the GMVAE**, the distributions parameters are learned during training.

→ **For the β-VAE**, we first encode the entire dataset:



How to obtain the distributions parameters?

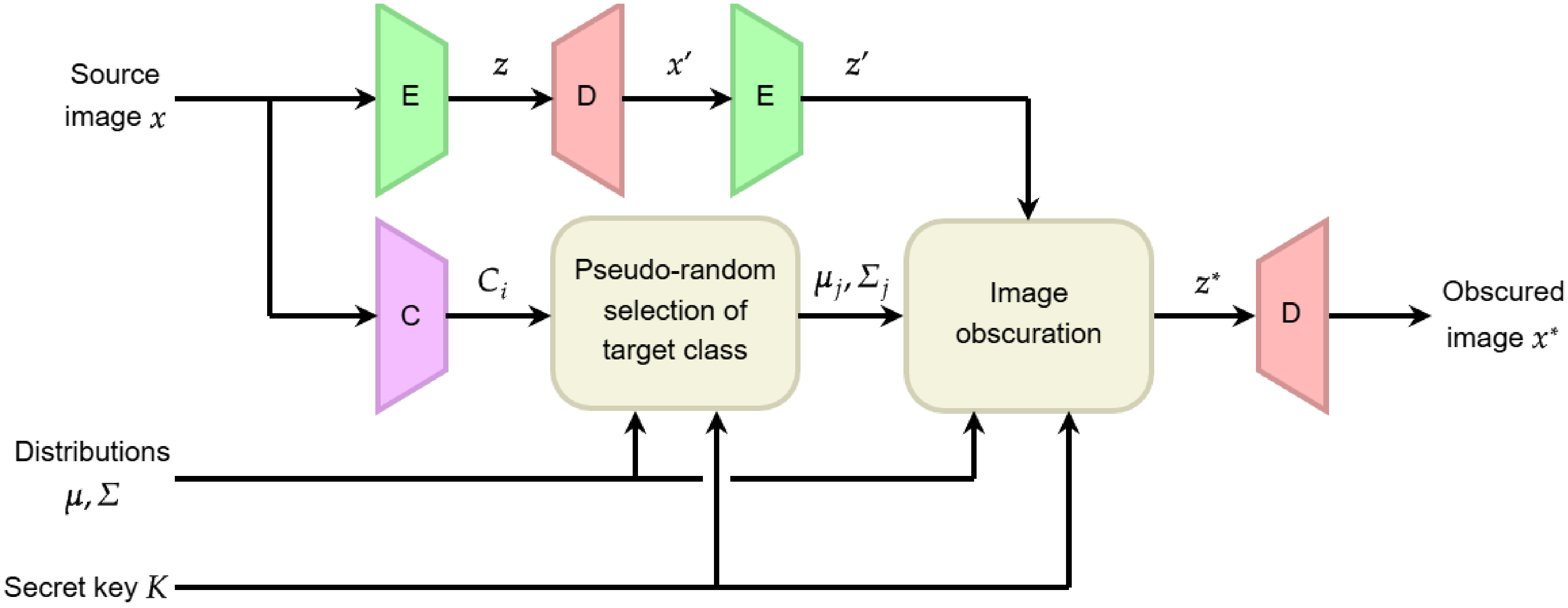
➔ From each latent variables for each class C_i, calculate the **mean**:

$$\boldsymbol{\mu}_i = \frac{1}{|\mathcal{X}_i|} \sum_{k=1}^{|\mathcal{X}_i|} \boldsymbol{z}_i^{\prime k}$$

➔ Then calculate the **covariance** for each class C_i:

$$\Sigma_i = \frac{1}{|\mathcal{X}_i|} \sum_{k=1}^{|\mathcal{X}_i|} (\boldsymbol{z}_i^{\prime k} - \boldsymbol{\mu}_i)(\boldsymbol{z}_i^{\prime k} - \boldsymbol{\mu}_i)^\top$$

Overview of the first 3 proposed obscuration methods

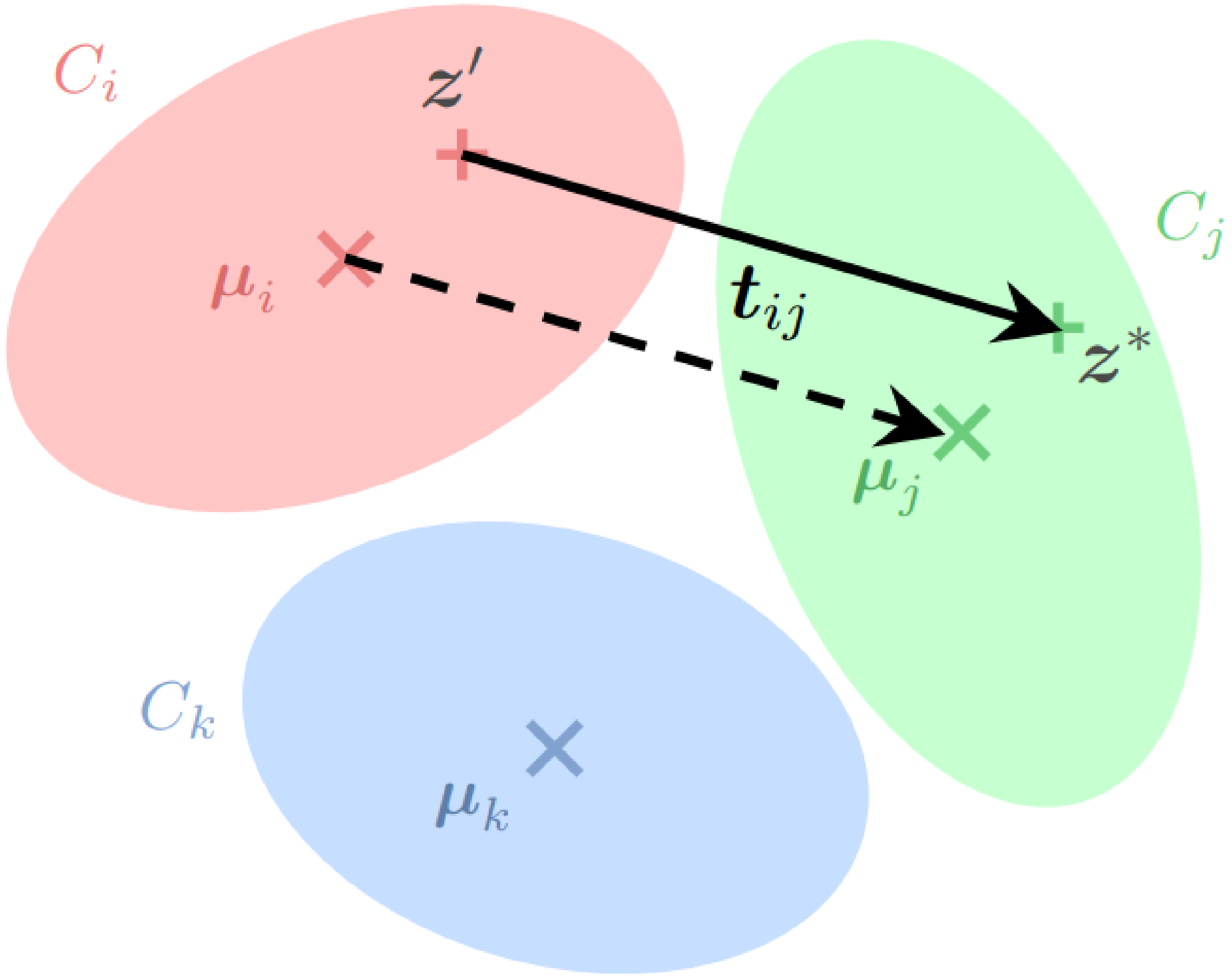


➔ Pseudo-random selection of target class:

$$u \sim \mathcal{U}(\{0, \ldots, m\}, K)$$
$$j = (i + u) \bmod m$$

➔ This method is **reversible**

Method 1: Translation



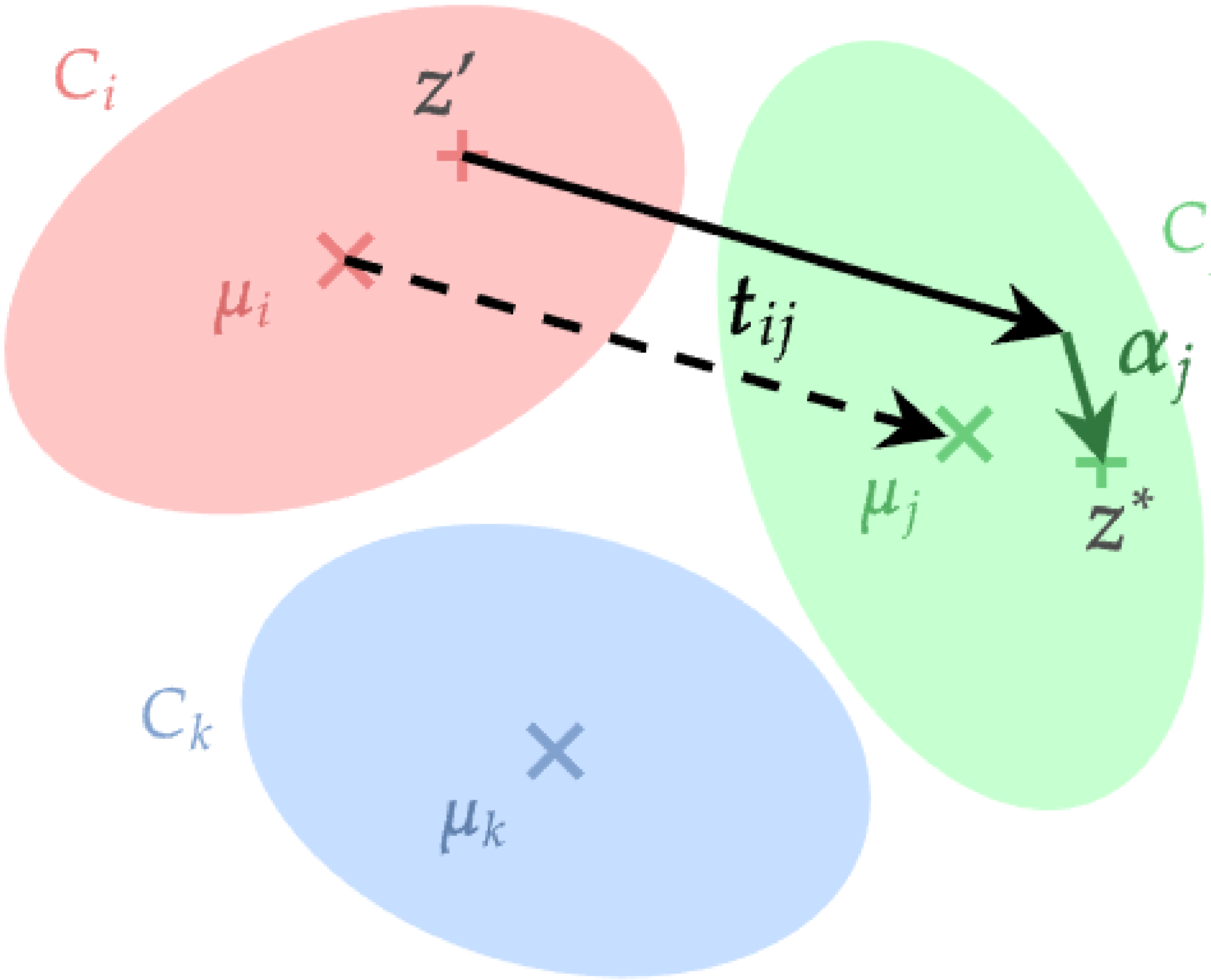
First, calculate the translation vector:

$$\boldsymbol{t}_{ij} = \boldsymbol{\mu}_j - \boldsymbol{\mu}_i$$

Then apply the translation from class C_i to class C_j:

$$\boldsymbol{z}^* = \boldsymbol{z}' + \boldsymbol{t}_{ij}$$

Method 2: Perturbed translation



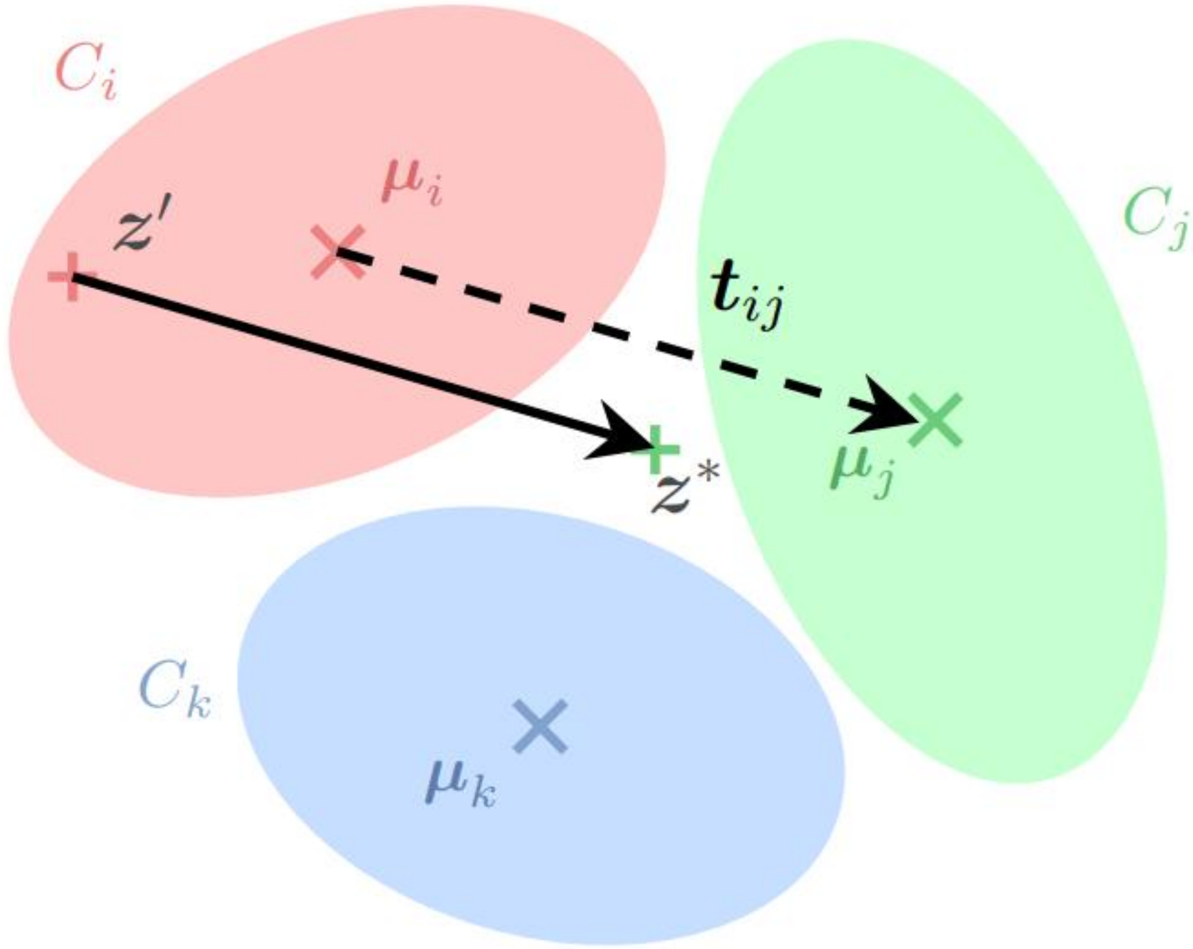
First, sample the perturbation from the target class variance.

$$\boldsymbol{\alpha}_j \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_j^{\mathbf{2}}, K)$$

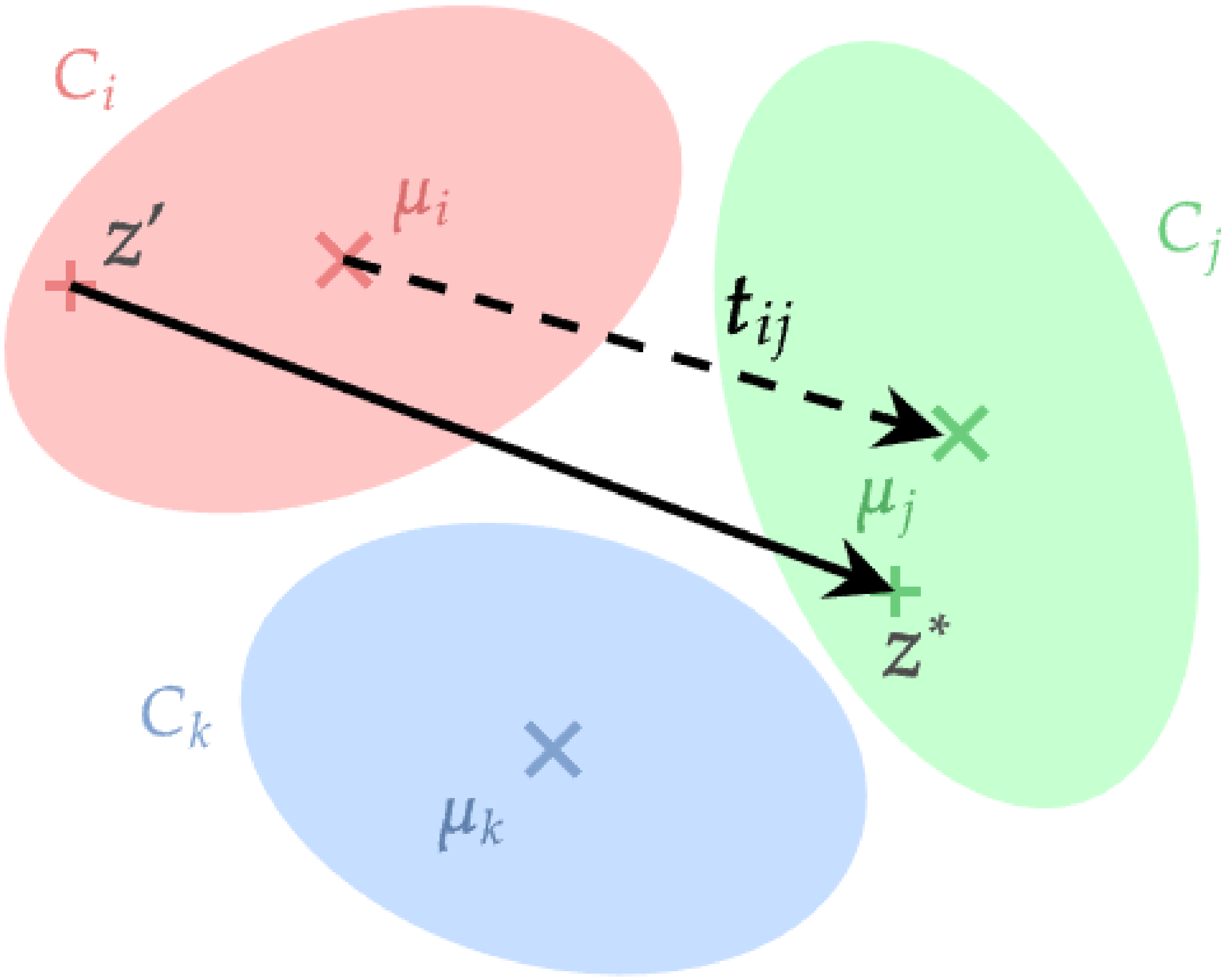
Then, apply the perturbed translation

$$\boldsymbol{z}^* = \boldsymbol{z}' + \boldsymbol{t}_{ij} + \boldsymbol{\alpha}_j$$

A case where a simple translation fails



Method 3: Multivariate transformation



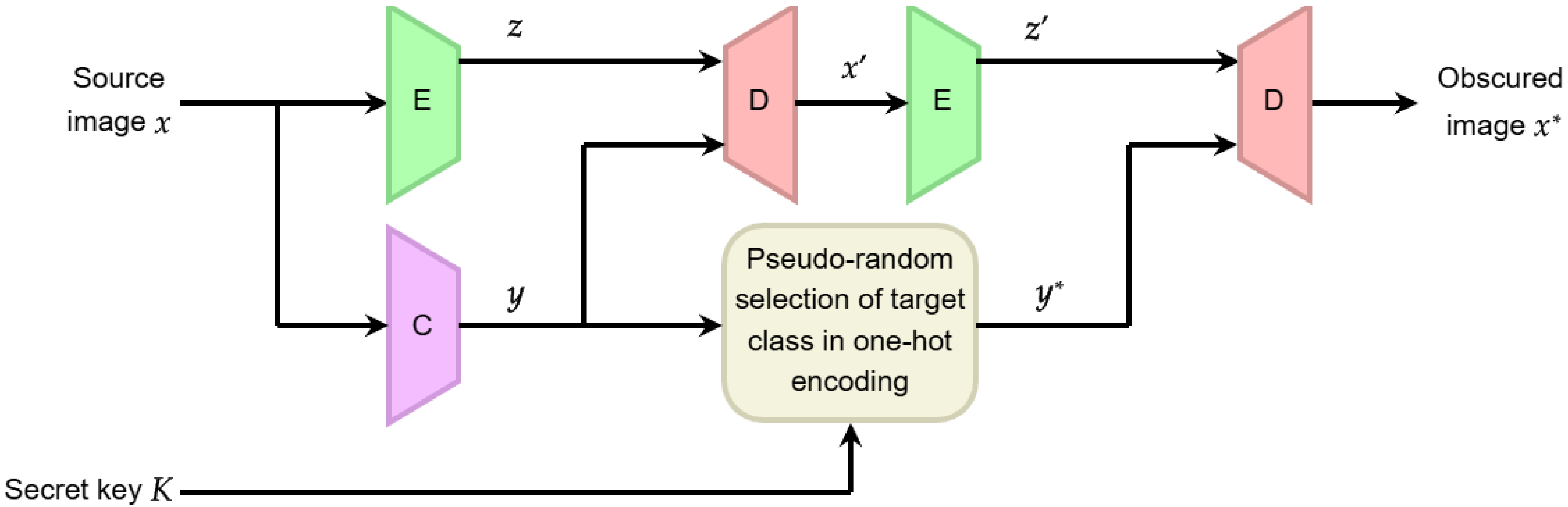
First, calculate the Cholesky factor for both classes:

$$\Sigma_i = L_i L_i^\top$$

Apply the transformation from class C_i to class C_j:

$$\boldsymbol{z}^* = \boldsymbol{\mu}_j + L_j L_i^{-1}(\boldsymbol{z}' - \boldsymbol{\mu}_i)$$

Method 4: Overview of the obscuration method via conditioning



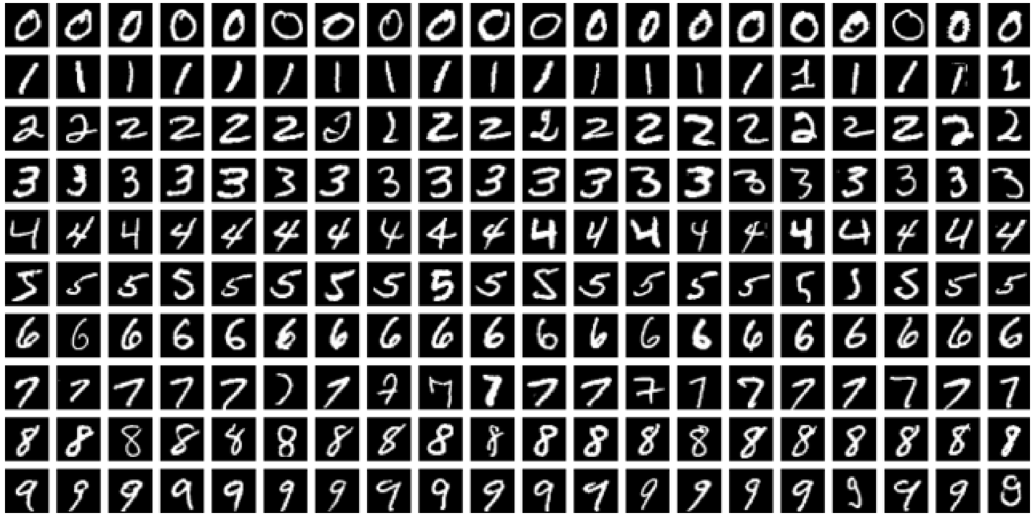
➔ For simplicity, assume that the classifier outputs in **one-hot encoding**

e.g. $C_3 = [0,0,0,1,0,0,0,0,0,0]$

Summary

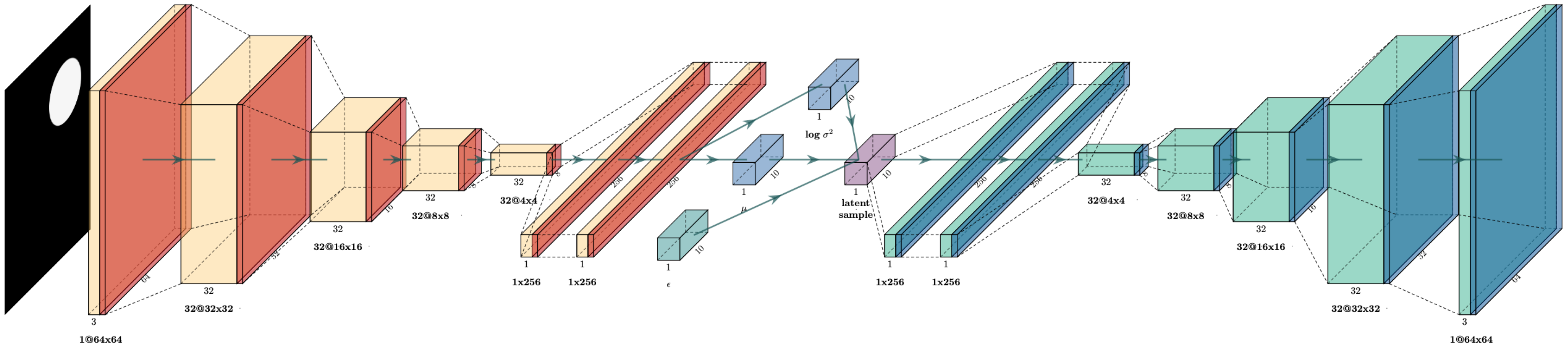
Dataset

- ➔ Dataset of handwritten digits **MNIST** (between 0 and 9)
- ➔ **60 000** training images
- ➔ **10 000** testing images



📖 **Yann LeCun, Corinna Cortes, Christopher J.C. Burges**
MNIST (Modified National Institute of Standards and Technology) handwritten digits database
http://yann.lecun.com/exdb/mnist/

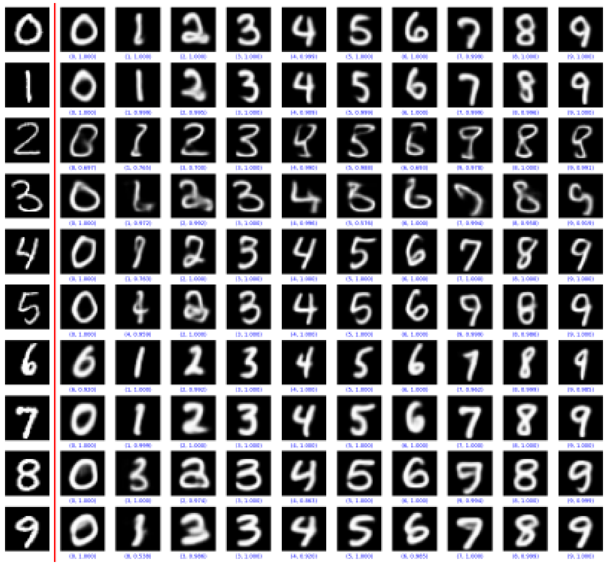
VAE model details



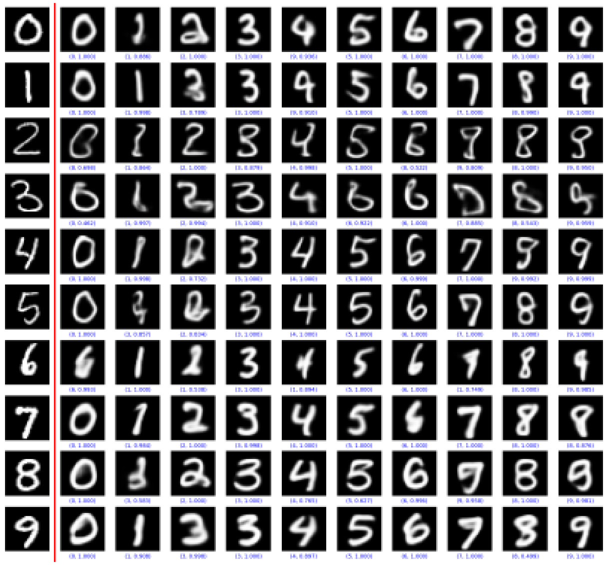
- ➔ Latent space of **128 dimensions**
- ➔ Trained for **5 epochs**
- ➔ Regularisation factor **β = 4.5**

📖 **Christopher P. Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, Alexander Lerchner**
Understanding disentangling in β-VAE
31st Conference on Neural Information Processing Systems (NIPS), 2017

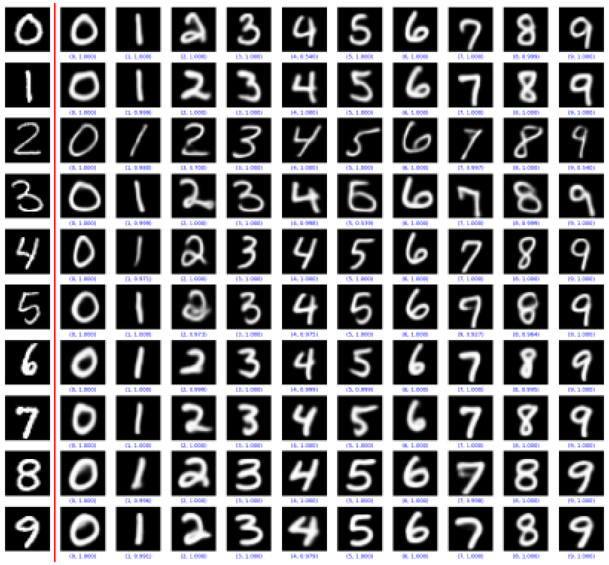
Obscuration results of our 4 obscuration methods



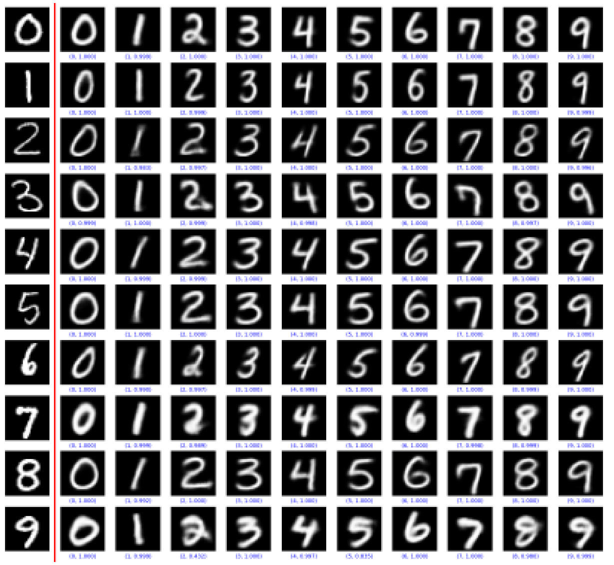
Translation



Perturbed translation

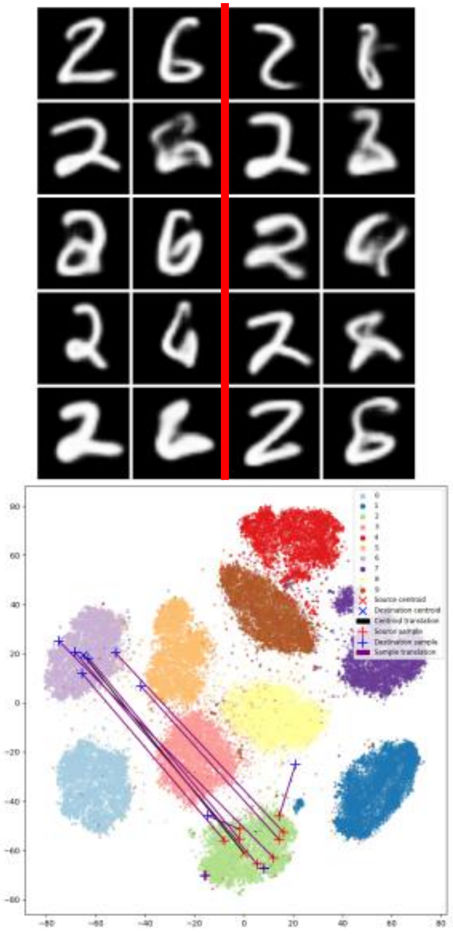


Multivariate transformation

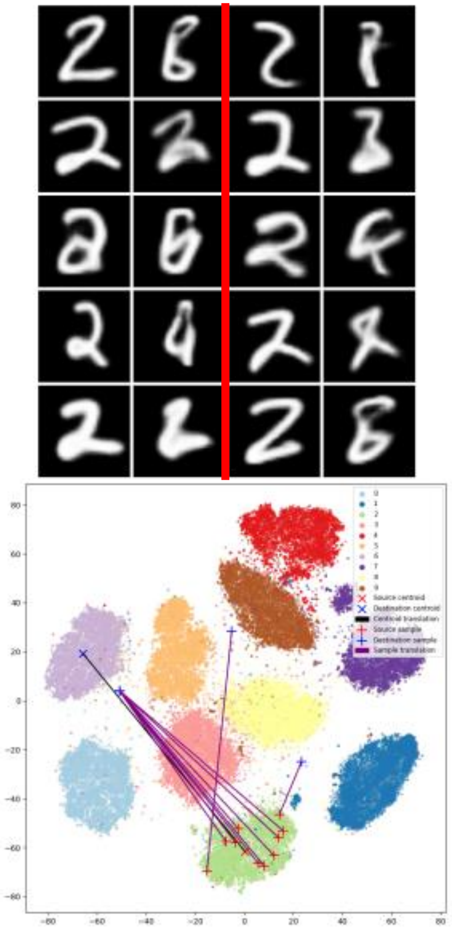


Conditioning

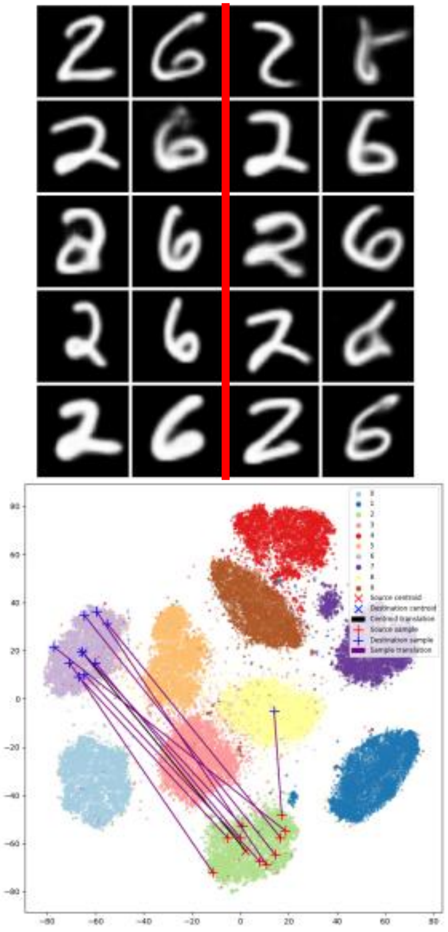
A link between perceptual quality and 2D projection



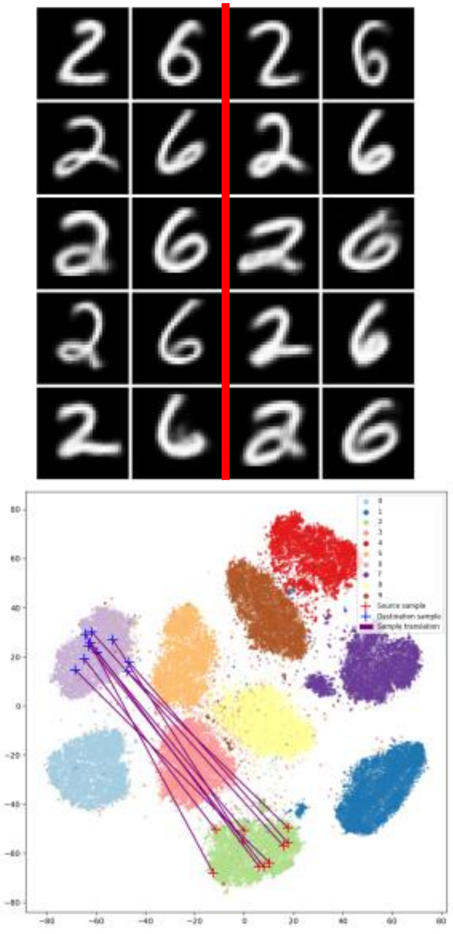
Translation



Perturbed translation

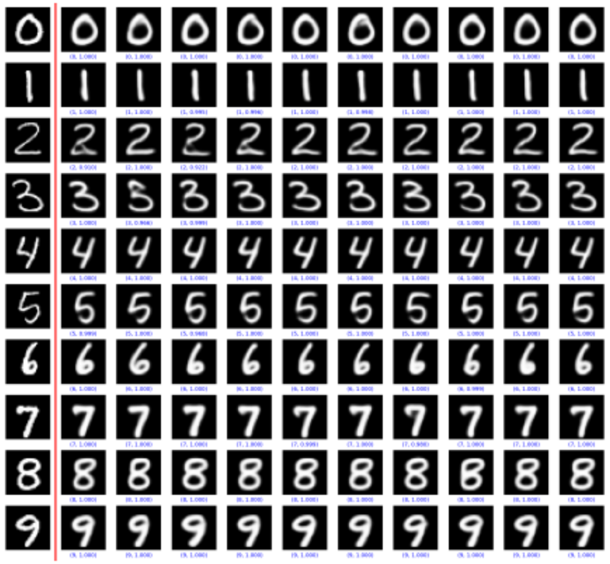


Multivariate transformation

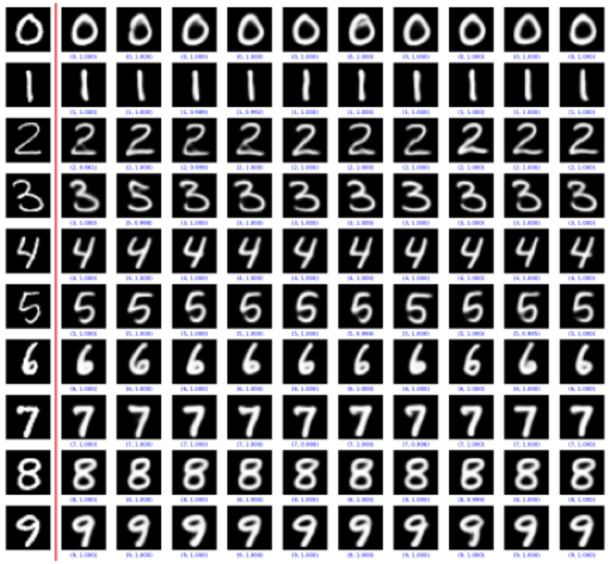


Conditioning

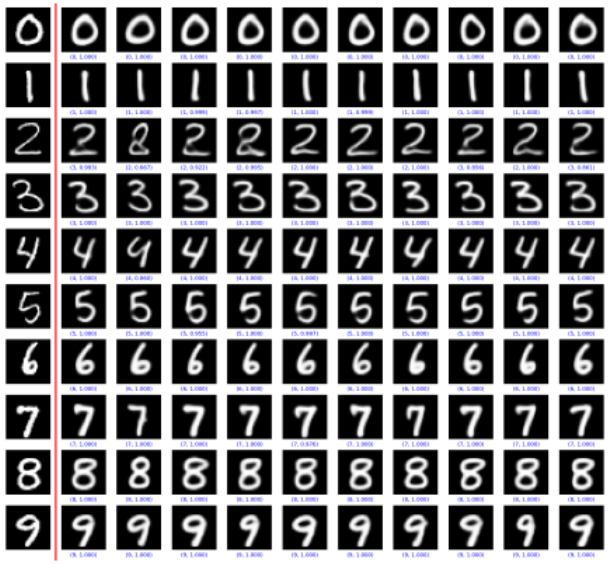
Reconstruction results of our 4 obscuration methods



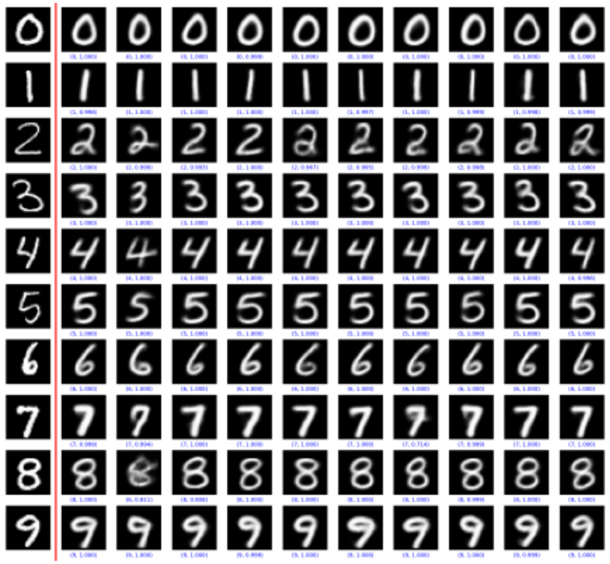
Translation



Perturbed translation

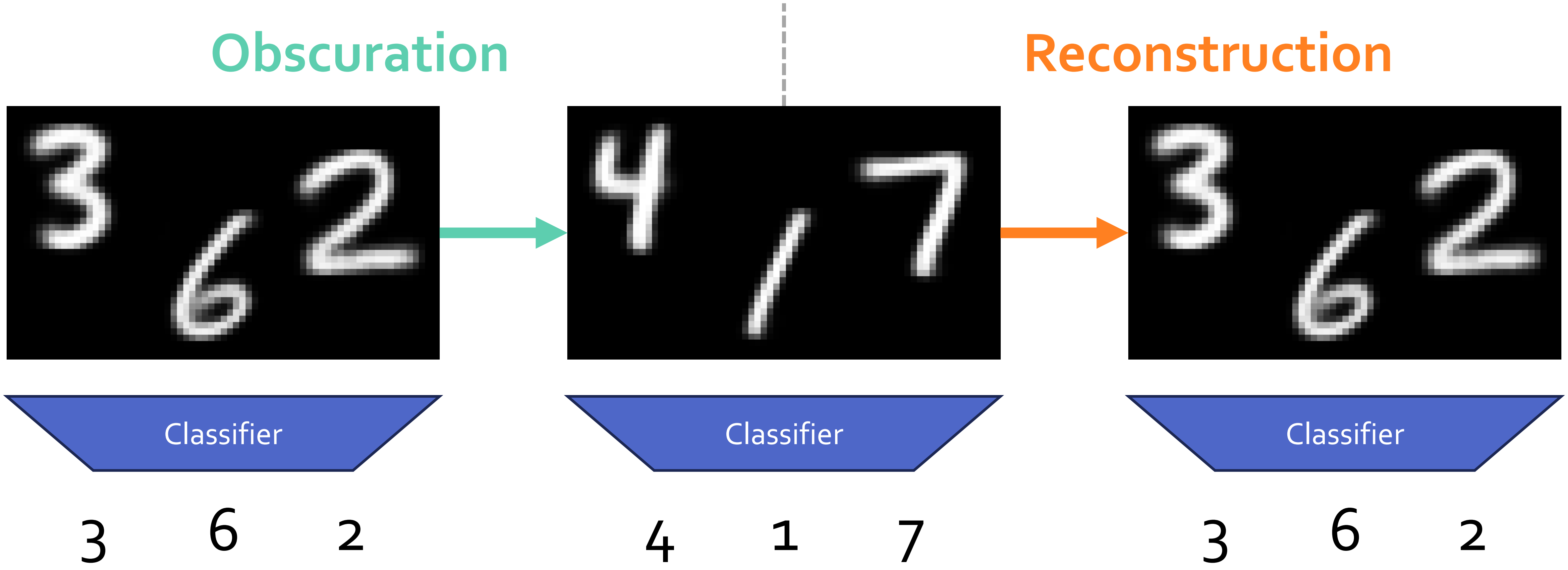


Multivariate transformation



Conditioning

Evaluation of the obscuration methods



Neural Network-based classifier model details (CNN classifier)

→ Trained for **20 epochs**

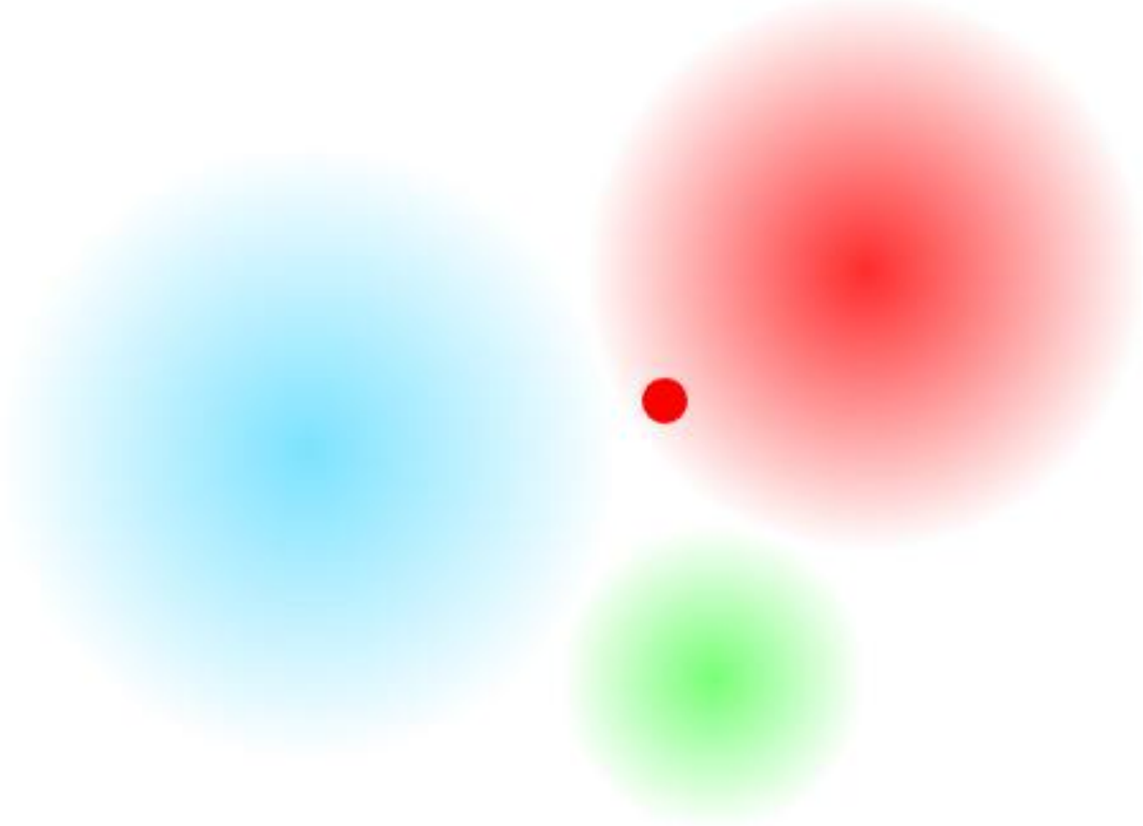
→ **99.26%** of accuracy on MNIST

→ **99.29%** of certainty on MNIST

Layer	Activation	Dimensions
Input x		$1 \times 28 \times 28$
64 convolutions 3×3		$64 \times 28 \times 28$
Max Pooling	ReLU	$64 \times 14 \times 14$
128 convolutions 3×3		$128 \times 14 \times 14$
Max Pooling	ReLU	$128 \times 7 \times 7$
Flatten		6272
Dense	ReLU	256
Dropout 50%		256
Dense	Softmax	10
Output $\hat{\boldsymbol{y}}$		10

Quadratic classifier model details (QDA Classifier)

- → Determines the most likely distribution the vector is in
- → **96.72%** of accuracy on MNIST



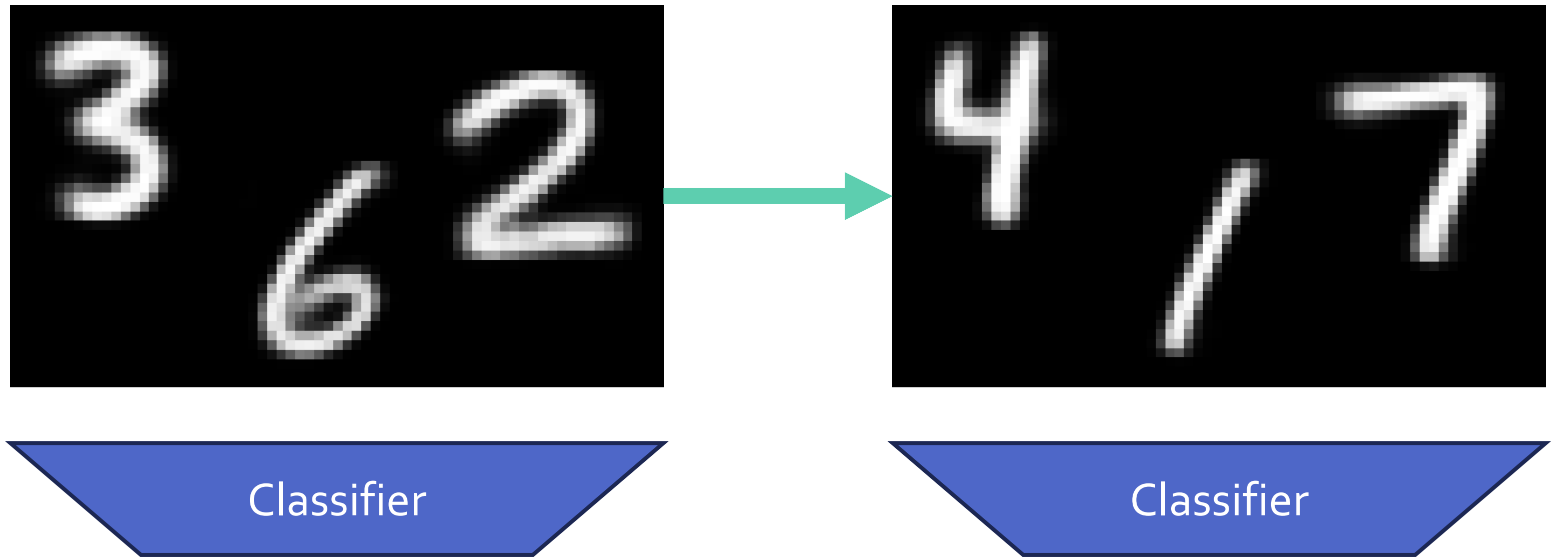
Evaluation of obscuration and reconstruction

Method		Translation	Perturbed translation	Multivariate transform	Conditioning
Obscuration	**CNN**	86.16%	78.04%	95.20%	**99.08%**
	QDA	56.20%	11.09%	**99.46%**	N/A
Reconstruction	**CNN**	97.47%	96.91%	96.38%	**99.72%**
	QDA	88.77%	11.63%	**99.26%**	N/A

Summary

Detection of the obscuration process

Obscuration



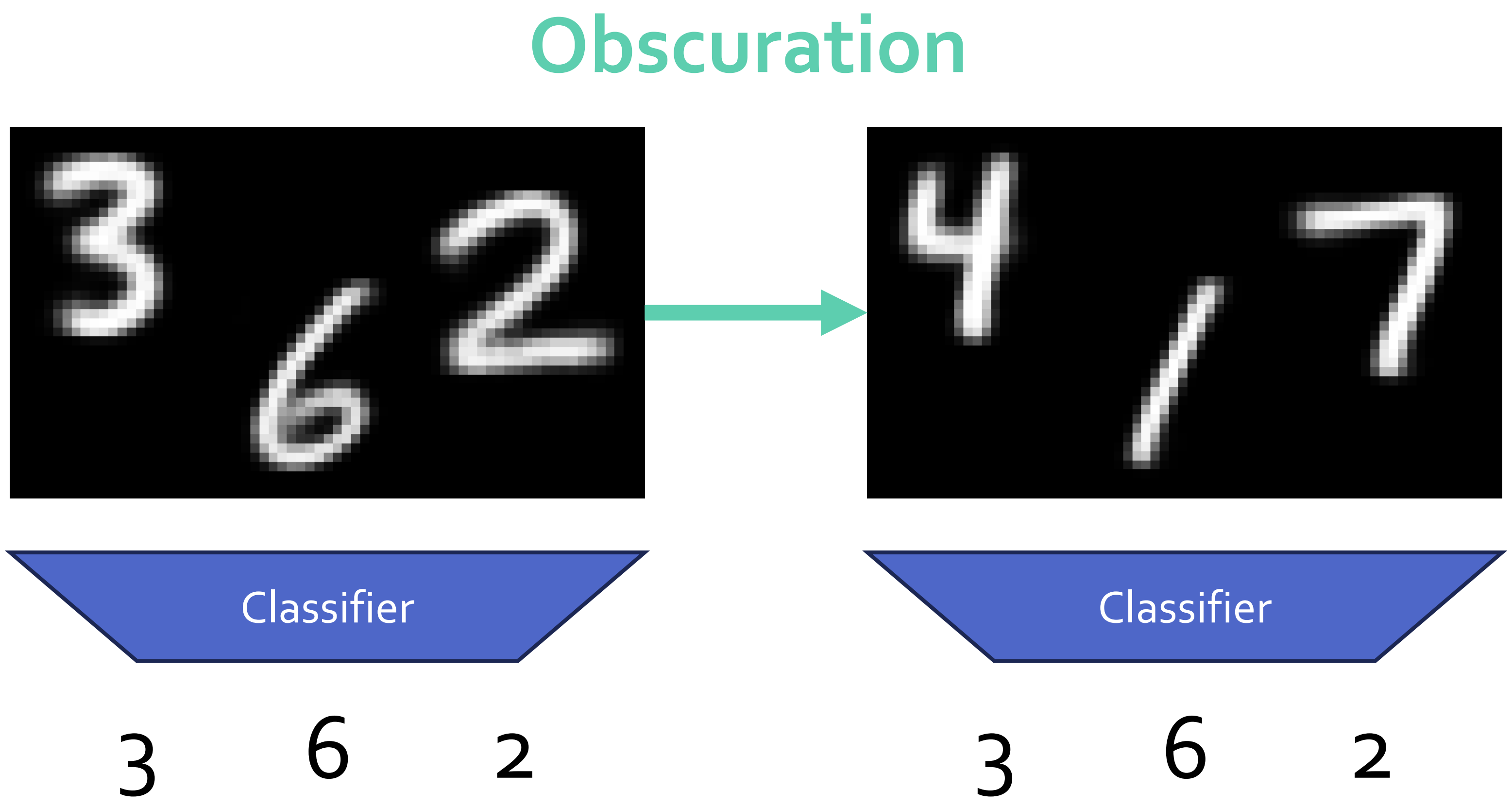
N N N

Original

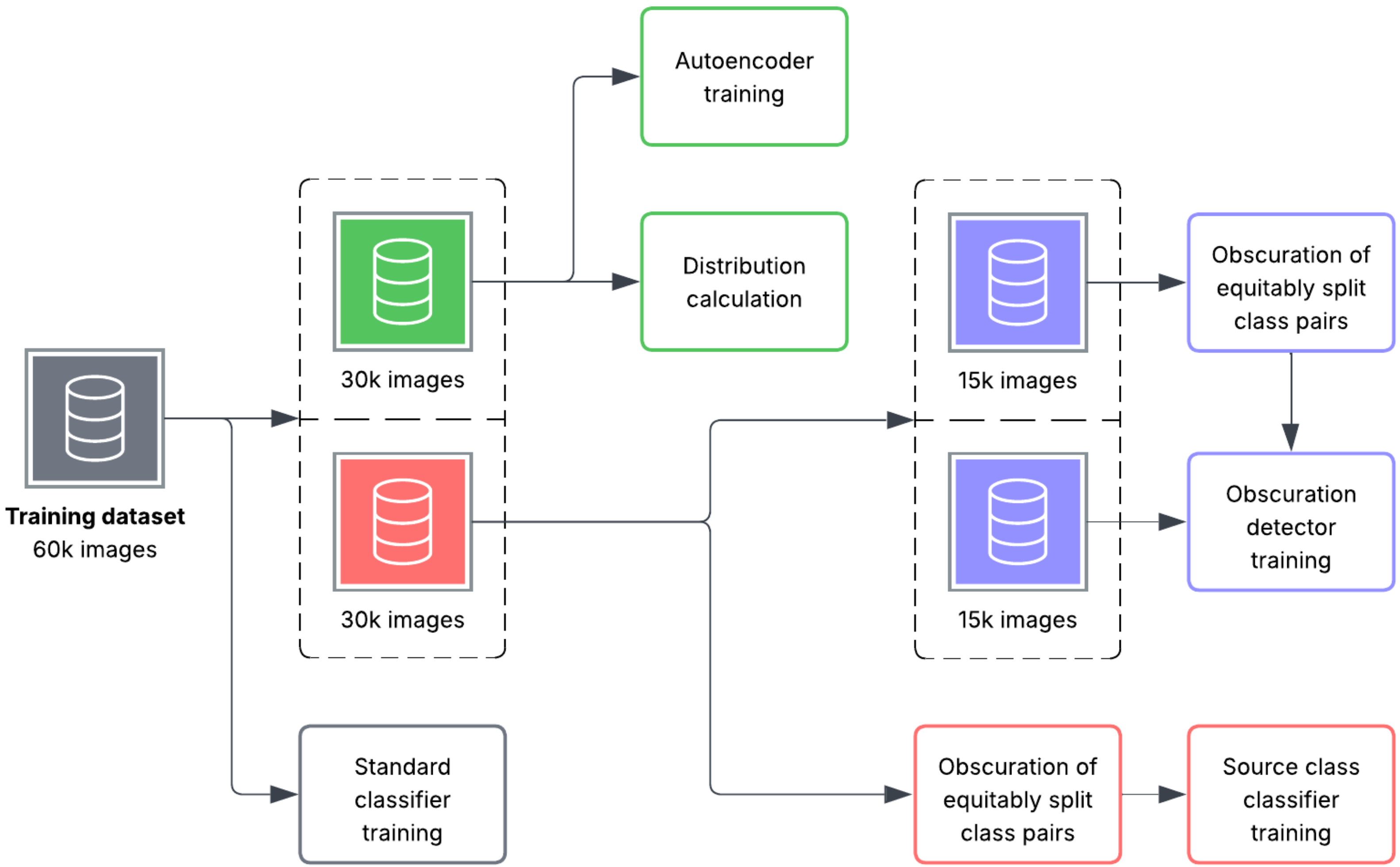
P P P

Obscured

Source class detection in obscured images



Dataset partition strategy for training



Evaluation of obscuration detection

Detection of obscuration	Translation	Perturbed translation	Multivariate transform	Conditioning
Original images	99.71%	99.61%	99.35%	**99.18%**
Reencoded images	81.14%	86.06%	76.32%	**63.33%**

Evaluation of source class detection

Source class detection	Translation	Perturbed translation	Multivariate transform	Conditioning
Standard classifier	11.34%	12.62%	**9.86%**	10.19%
Specialized classifier	58.98%	41.51%	**14.56%**	20.10%

Conclusion

→ **4 approches to image content obscuration:**

- → Efficient *(95-99% classification accuracy of obscuration)*
- → Invisible *(the style is kept during the obscuration process)*
- → Reversible *(95-99% classification accuracy of reconstruction)*

→ **Attack of our methods:**

- → Detectable, but hardly *(63-76% classification accuracy)*
- → Robust against source class inference *(14-20% classification accuracy)*

Prospects

→ **Obscuration on larger datasets**: using larger datasets to capture more variance in data (SVHN, ImageNet, ...)

→ **Class-agnostic obscuration**: obscuring an image without class labels or class distributions

→ **Obscuration on more expressive multimodal distribution**: taking in account intra-class and inter-class variances

→ **Using more complex architectures**: Vision Transformers (ViT), diffusion models, etc.

The End

Thank you for listening

Do you have any questions?