Variational Mixture of HyperGenerators for Learning Distributions over Functions

Ignacio Peis

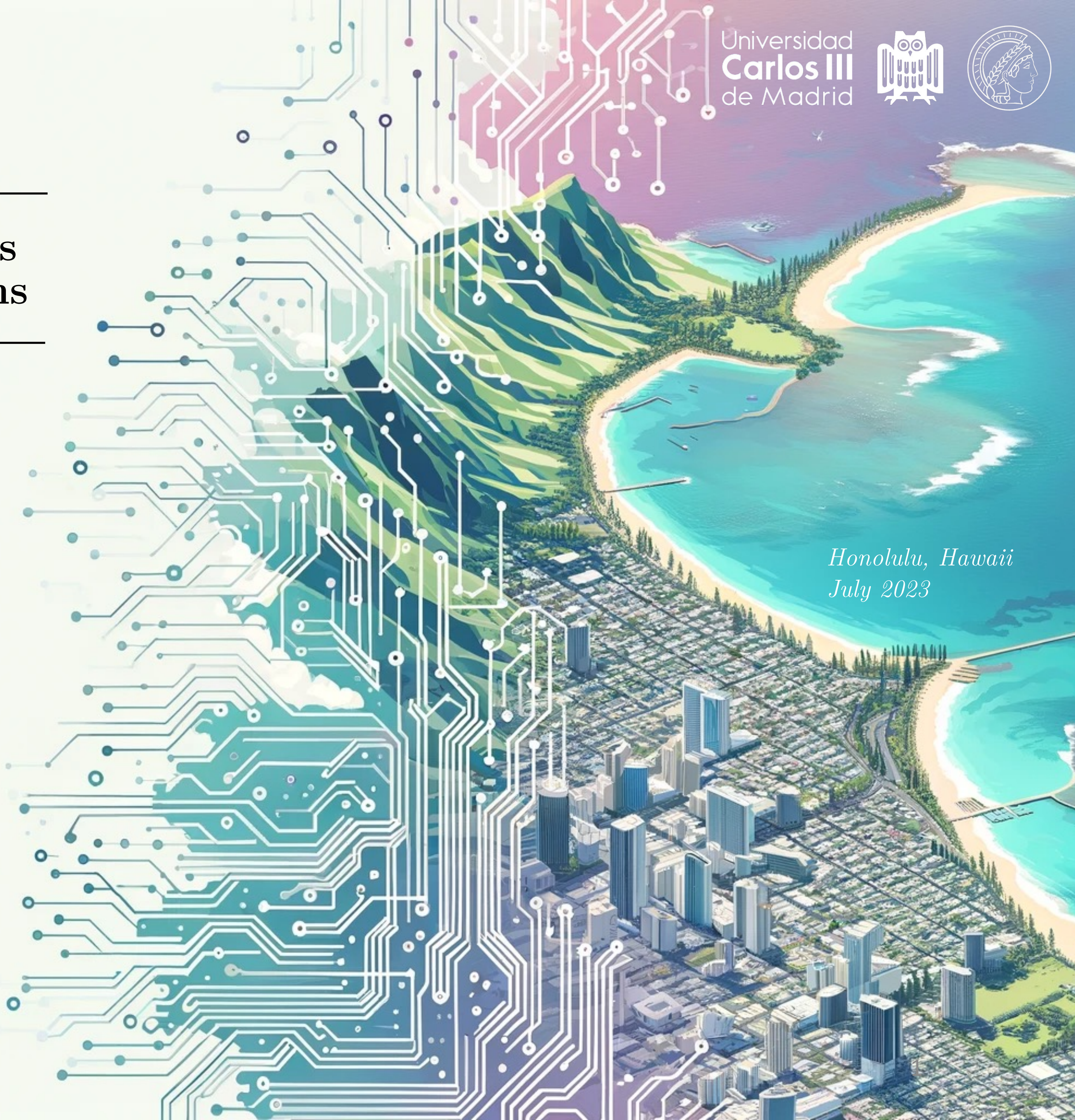
ipeaz@dtu.dk

Technical University of Denmark

DTU

ICML
International Conference
On Machine Learning

Andaluz.IA



Collaborators





Batuhan Koyuncu[1] Pablo Sánchez[1,2,3] Ignacio Peis[4] Pablo M. Olmos[5] Isabel Valera[1,6]

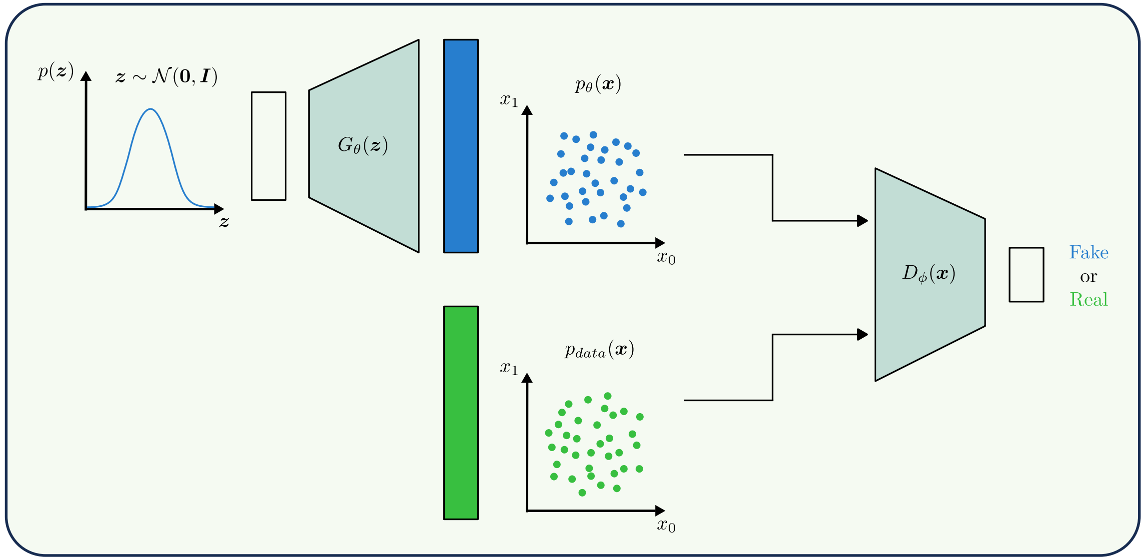
[Paper] [Code]

[1] University of Saarland
[2] Max Planck Institute for Intelligent Systems
[3] Sony AI
[4] Technical University of Denmark (DTU)
[4] Universidad Carlos III de Madrid
[5] Max Planck Institute for Software Systems

Deep Generative Models

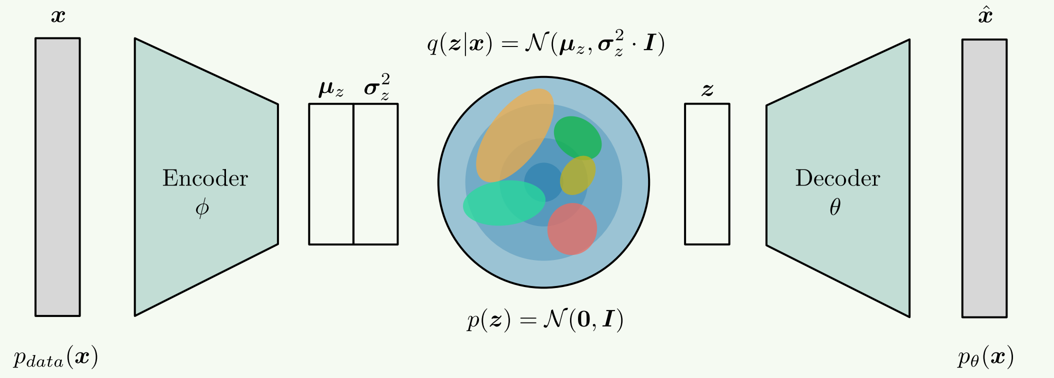
- Learning **probability distributions** on data using Deep Neural Networks.

Implicitly



Generative Adversarial Networks (GANs [1])

Explicitly



$$\log p(x) = \log \int p(x, z) dz \geq \mathbb{E}_{q(z|x)}[\log p(x|z)] - KL(q(z|x)||p(z))$$

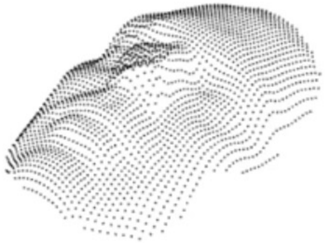
Variational Autoencoders (VAEs [2])
Denoising Diffusion Probabilistic Models (DDPMs [3])
Score-based models [4]
Energy-based models [5]

[1] (Goodfellow et al., 2014) [2] (Kingma et al., 2014) [3] (Ho et al., 2020) [4] (Song et al., 2019) [5] (Lecun et al., 2006)

Discretization of data

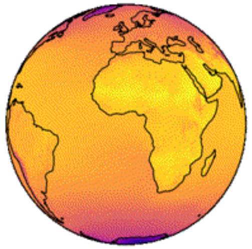
- We typically deal with discretized versions of data that are continuous in nature.

3D shapes



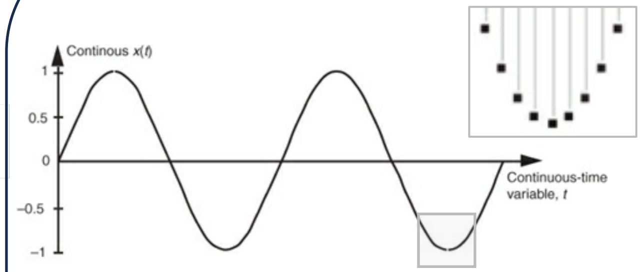
2D Images

Polar data



Spatial

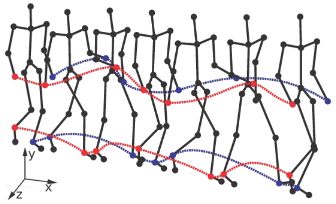
Signals



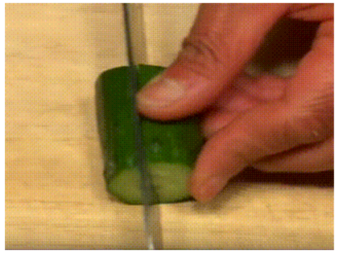
Audios

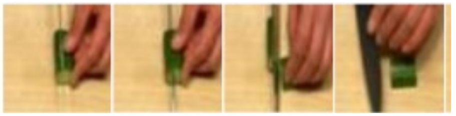
Temporal

Motion sequences



Videos



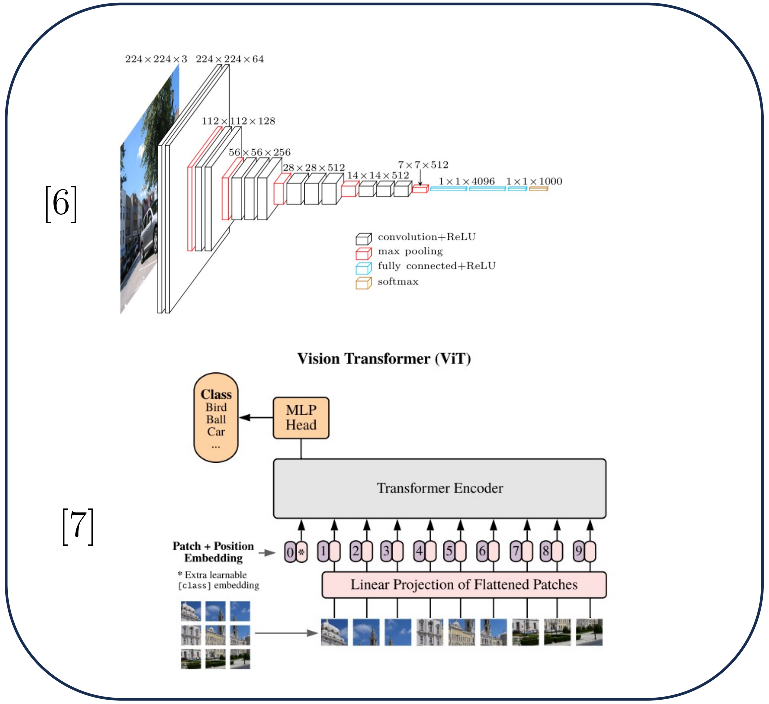


Spatio-temporal

NNs to exploit discretized data

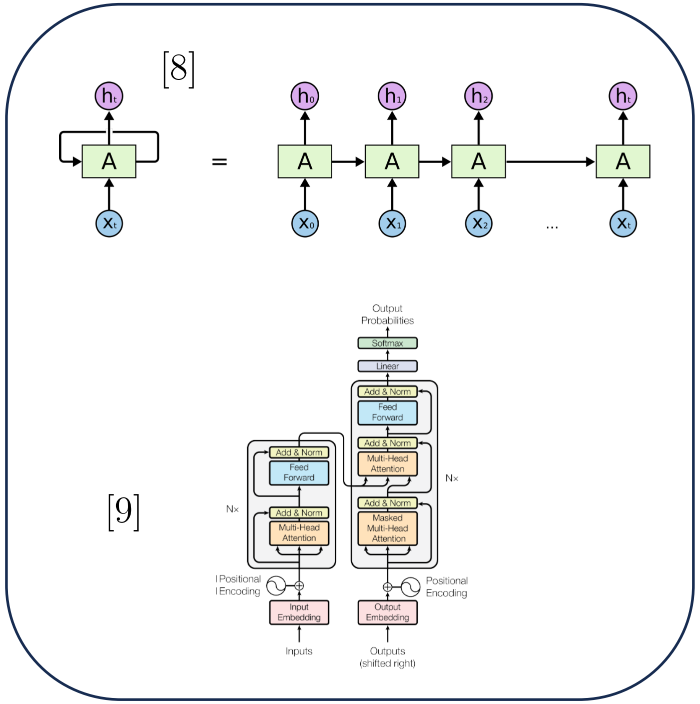
- DNNs are tailored to the data nature.

CNNs, Vision Transformers



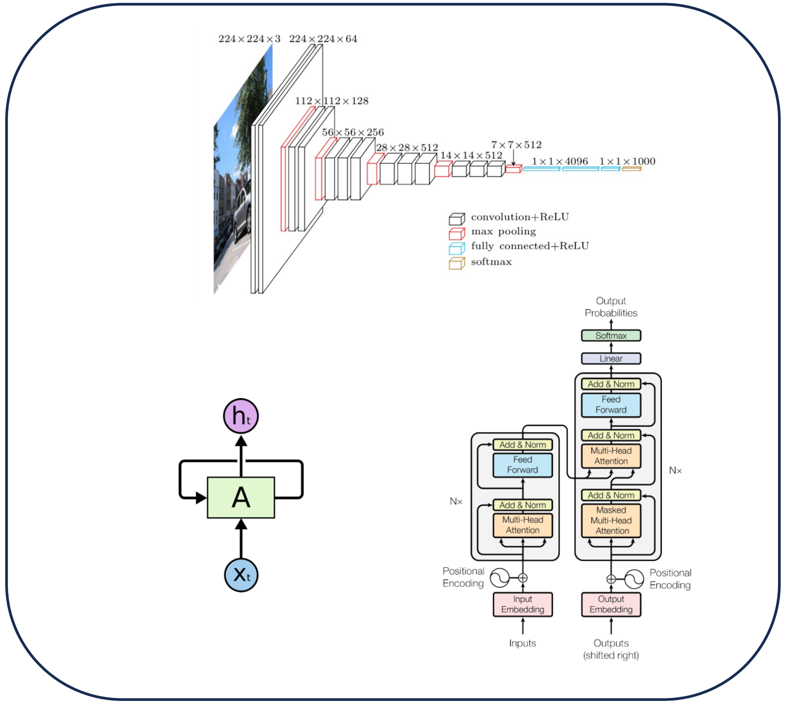
Spatial

RNNs, Transformers



Temporal

CNNs, RNNs, Transformers



Spatio-temporal

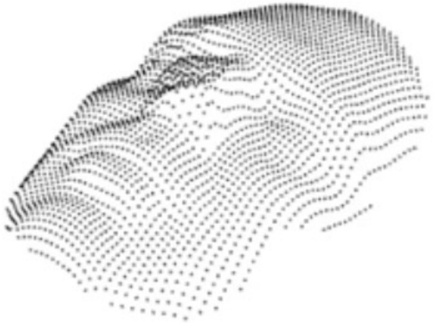
[6] (Simonyan et al., 2014) [7] (Dosovitskiy et al., 2020) [8] (Hochreiter et al., 1997) [9] (Vaswani et al., 2017)

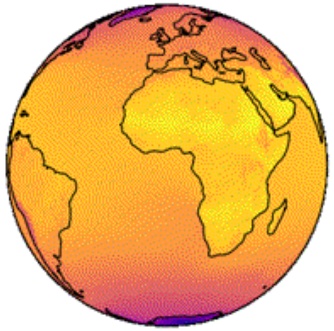
Real data is continuous in nature

- What if we approximate the underlying **continuous functions**?

$f : \mathbb{R}^2 \to \mathbb{R}^3, f(x_1, x_2) = (r, g, b)$

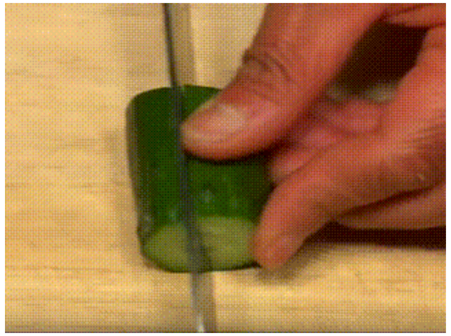
$f : \mathbb{R}^3 \to \{0, 1\}, f(x_1, x_2, x_3) = p$





$f : \mathbb{R}^2 \to \mathbb{R}, f(\varphi, \lambda) = T$

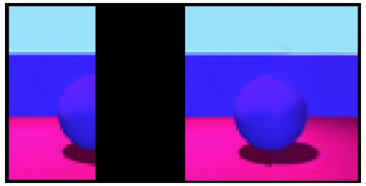
$f : \mathbb{R}^3 \to \mathbb{R}^3, f(x_1, x_2, t) = (r, g, b)$



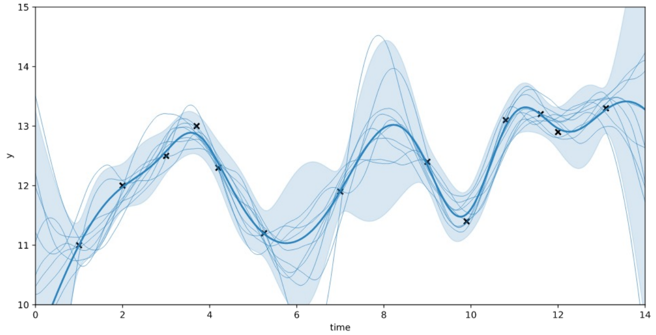
Real data is continuous in nature

- Learning the distribution of a function allows for **naturally handling**:

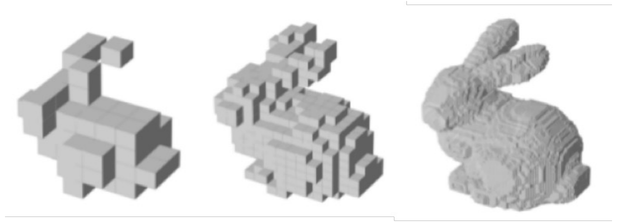
Inpainting



Outpainting



Conditional generation



Super-resolution

- We can use the same neural architecture independently of the data nature.
- Information to store will be independent of the data size.

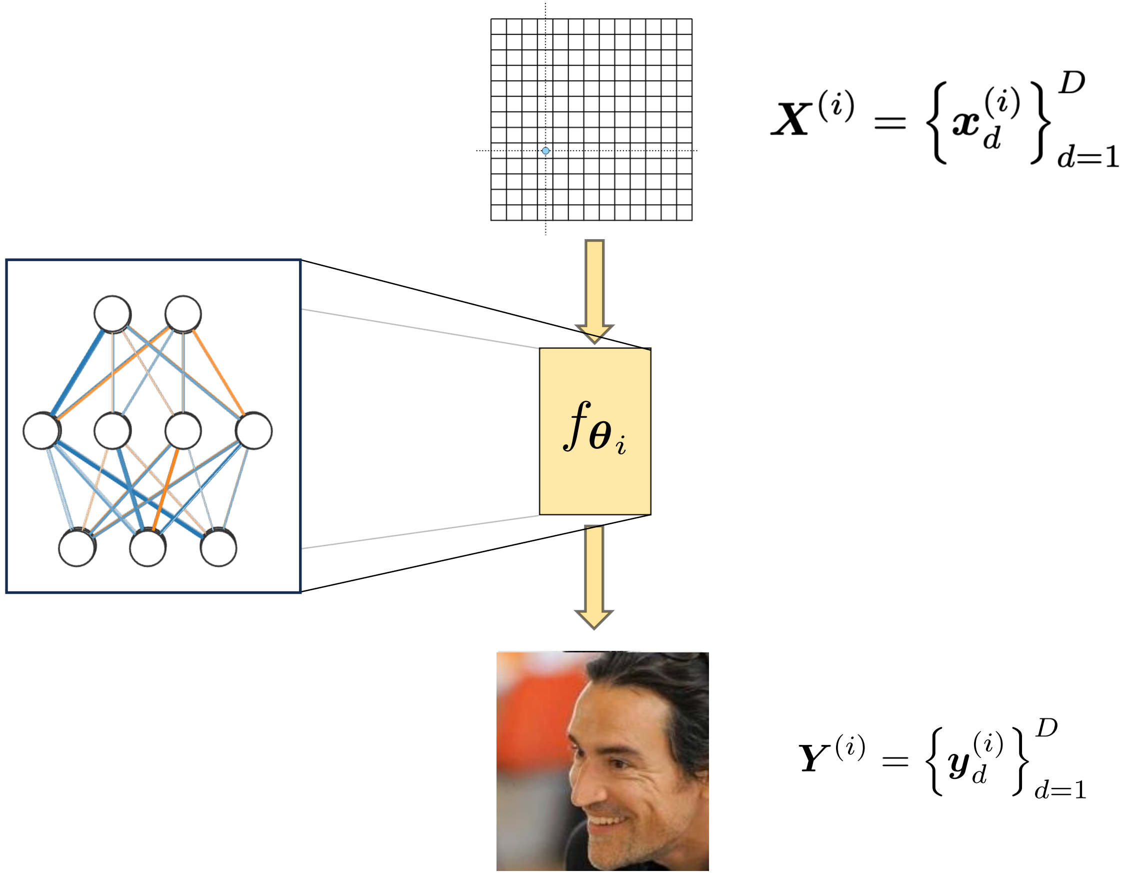
Implicit Neural Representations

- INRs [10-12] can approximate these functions.

How to efficiently learn unique INRs per datapoint?

↓

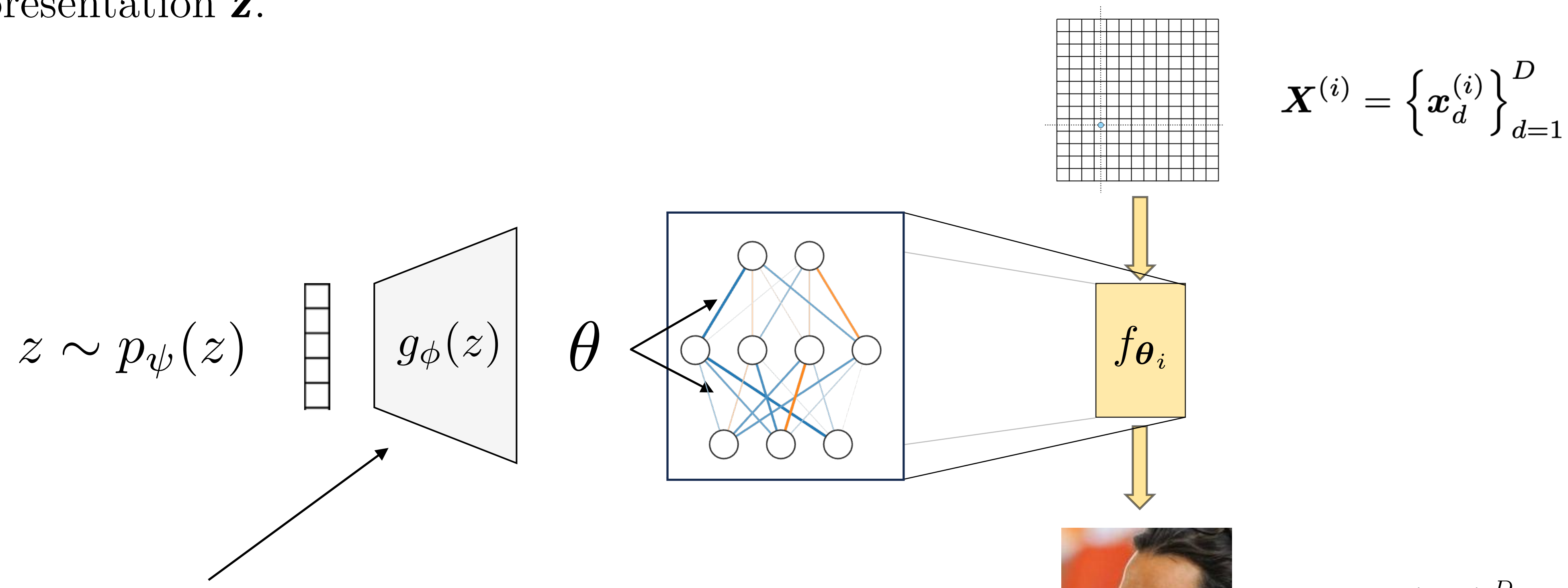
DGM that generate weights!

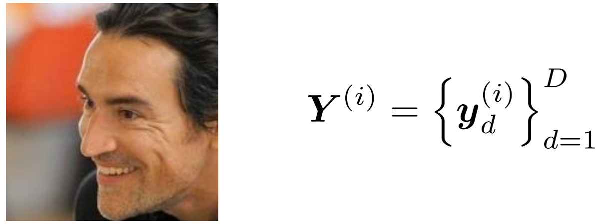


[10] (Sitzmann et al., 2020) [11] (Mescheder et al., 2019) [12] (Stanley et al., 2007)

Our proposed method

- Every set of weights and biases, $\boldsymbol{\theta}_i$, comes from a reduced latent representation $\boldsymbol{z}$.





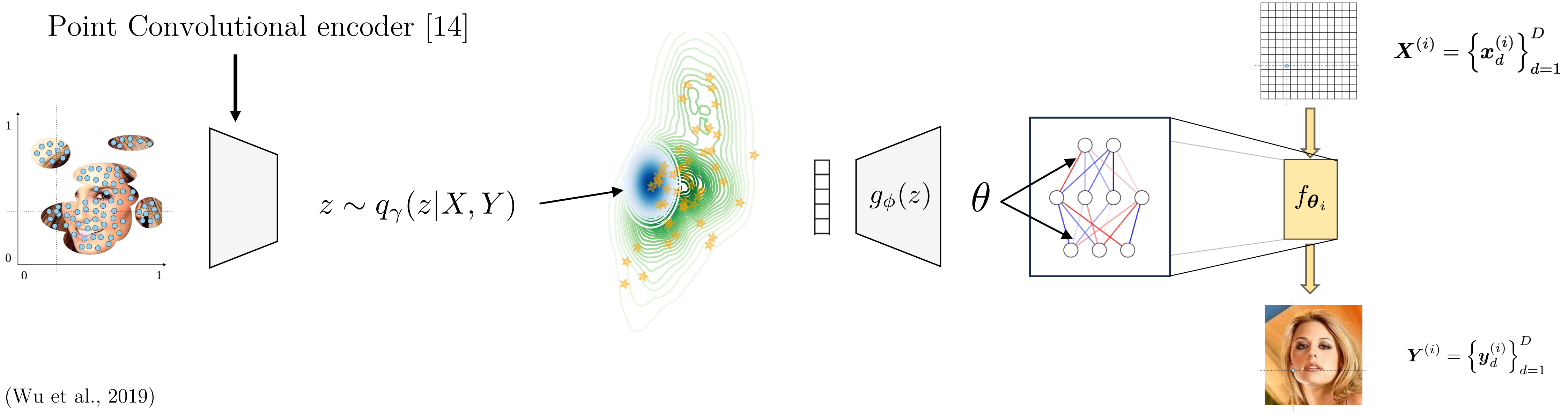
- A **hypernetwork** [13] converts $\boldsymbol{z}$ into weights and biases.

[13] (Ha et al., 2017)

Our proposed method

- To learn the parameters of our model, we opt by using **Amortized Variational Inference**, and optimize the following ELBO.

$$\max_{\phi,\psi,\gamma} \mathcal{L}(\phi,\psi,\gamma;\boldsymbol{Y},\boldsymbol{X}) = \max_{\phi,\gamma} \mathbb{E}_{q_{\boldsymbol{\gamma}}(\boldsymbol{z}|\boldsymbol{Y},\boldsymbol{X})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{z})\right] - D_{KL}\left(q_\gamma\left(\boldsymbol{z}|\boldsymbol{Y},\boldsymbol{X}\right)\|p_\psi\left(\boldsymbol{z}\right)\right)$$



[14] (Wu et al., 2019)

Our proposed method

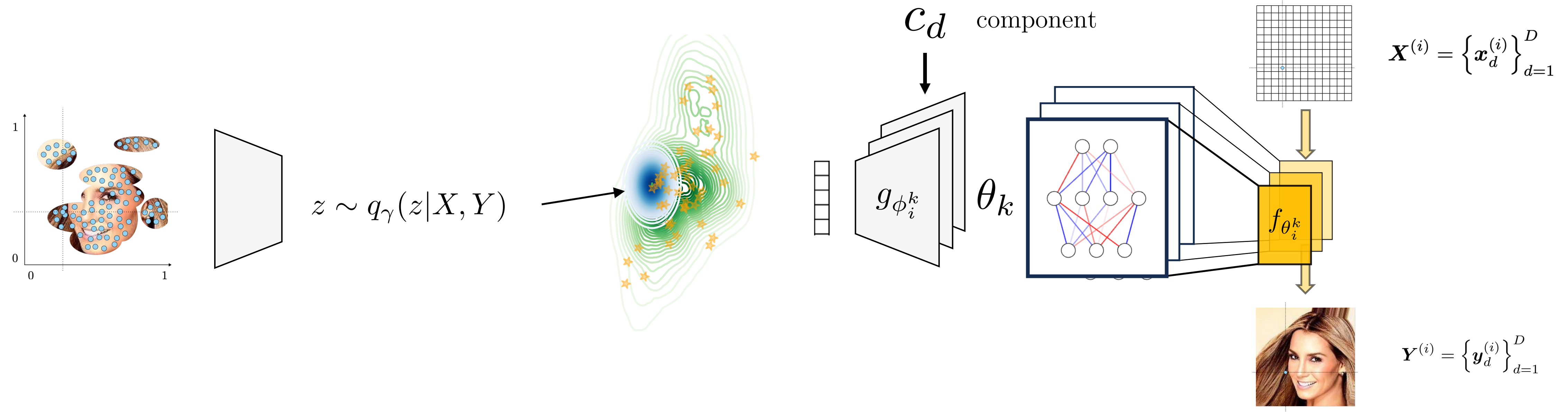
- We incorporate a **Mixture of HyperGenerators** for increased flexibility.

$$\mathcal{L}(\boldsymbol{Y},\boldsymbol{X};\psi,\phi,\boldsymbol{\gamma}) = \sum_{d=1}^{D} \mathbb{E}_{q_{\gamma_z}(\boldsymbol{z}|\boldsymbol{Y},\boldsymbol{X})}\left[\sum_{k=1}^{K} \log p_{\boldsymbol{\theta}_k}(\boldsymbol{y}_d \mid \boldsymbol{x}_d)\cdot \pi_{dk}\right] - D_{KL}\left(q_{\gamma_z}(\boldsymbol{z}\mid \boldsymbol{X},\boldsymbol{Y}) \| p_{\psi_z}(\boldsymbol{z})\right)$$
$$- D_{KL}\left(q_{\gamma_c}(\boldsymbol{C}\mid \boldsymbol{z},\boldsymbol{X},\boldsymbol{Y}) \| p_{\psi_c}(\boldsymbol{C}\mid \boldsymbol{z},\boldsymbol{X})\right)$$

Reconstruction

KL of the continuous latent variable

KL of the discrete latent variable



Our proposed method

- We incorporate a **Mixture of HyperGenerators** for increased flexibility.

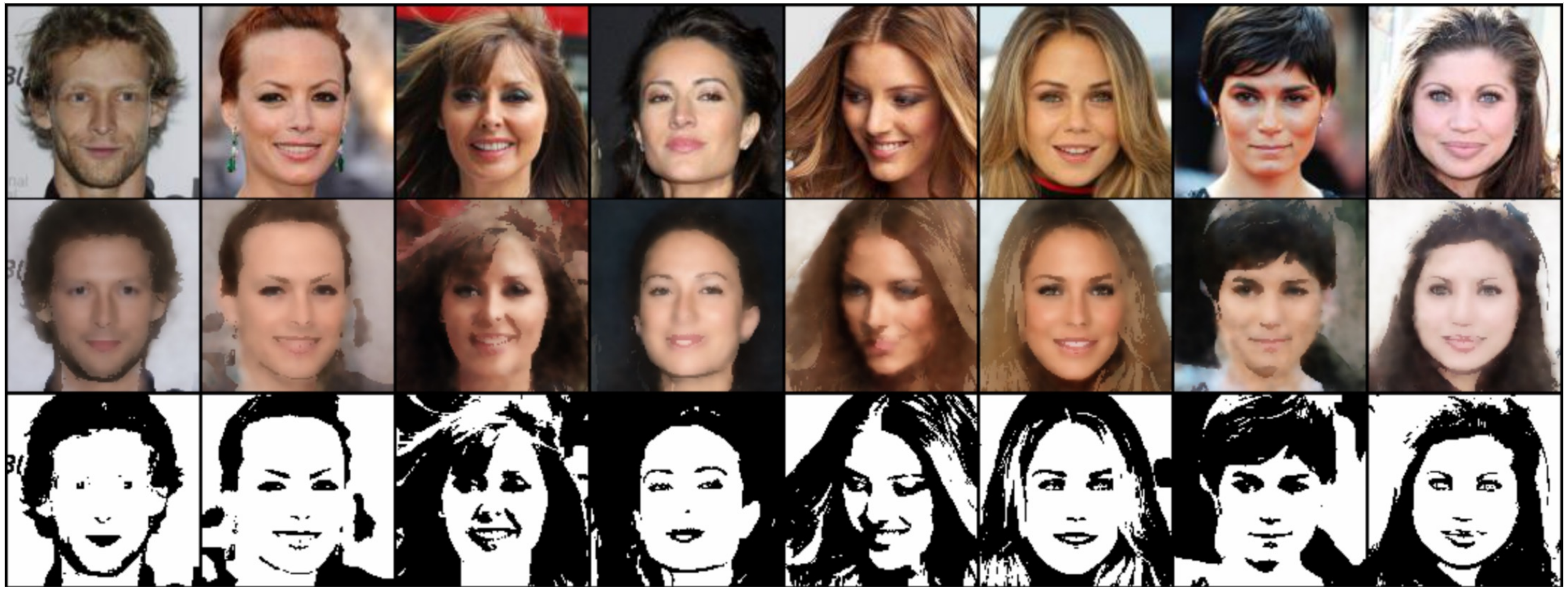
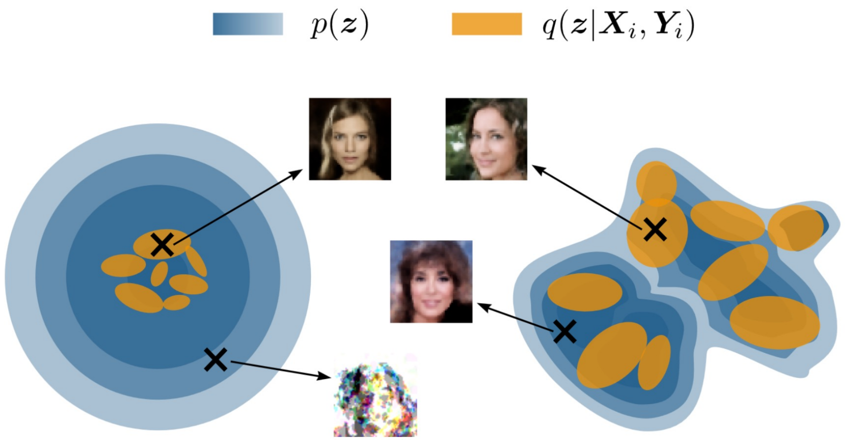
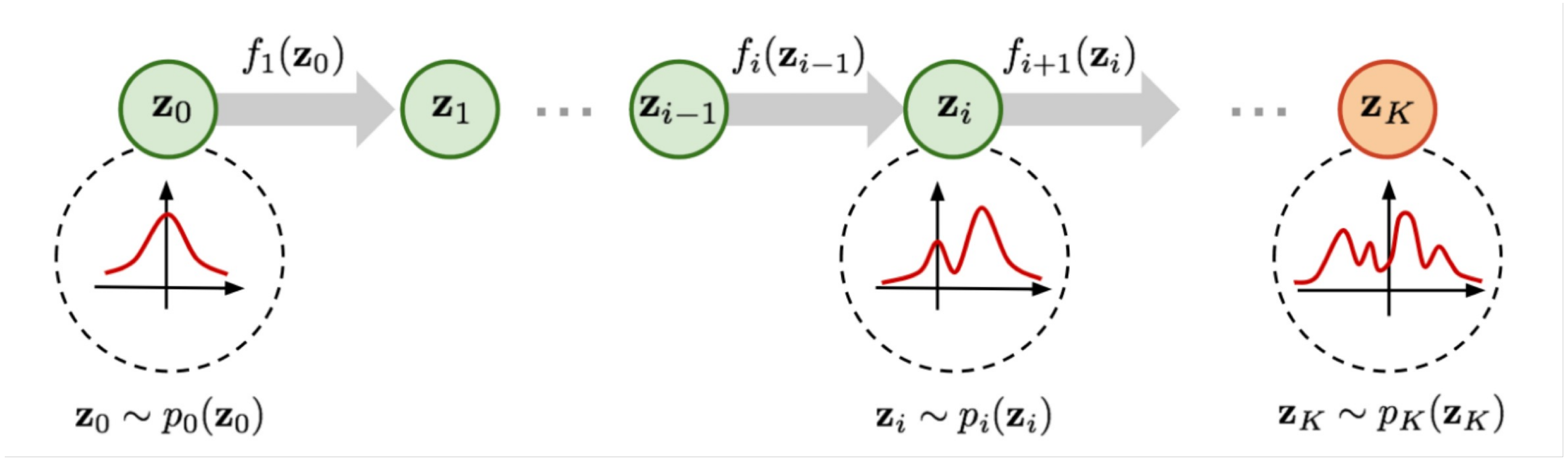


Image Reconstruction with Mixture of HyperGenerators

Our proposed method



- To alleviate the **holes** problem, similarly like Latent Diffusion models [15], we learn the prior as a planar Flow [16].

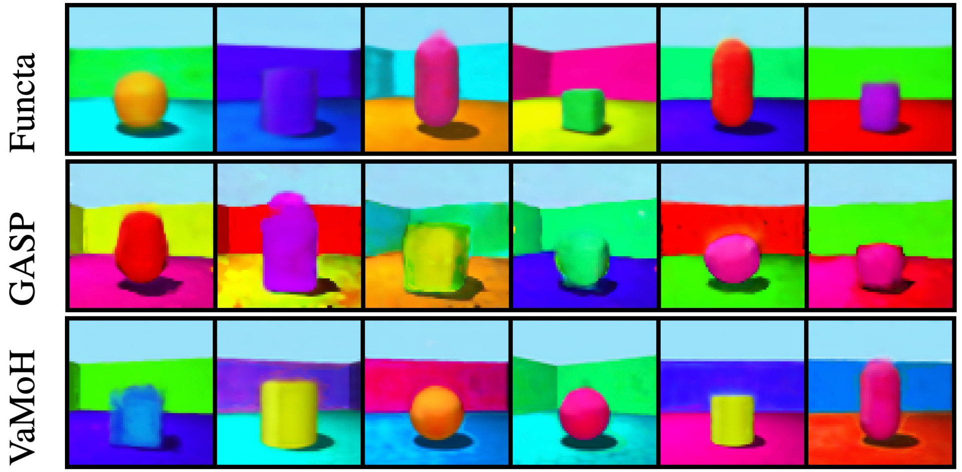


$$\boldsymbol{z}^{(i)} \sim p_\psi(z)$$

[15] (Rombach et al., 2022) [16] (Rezende et al., 2015)

Results

- We achieve **comparable sampling quality** and diversity wrt baselines.

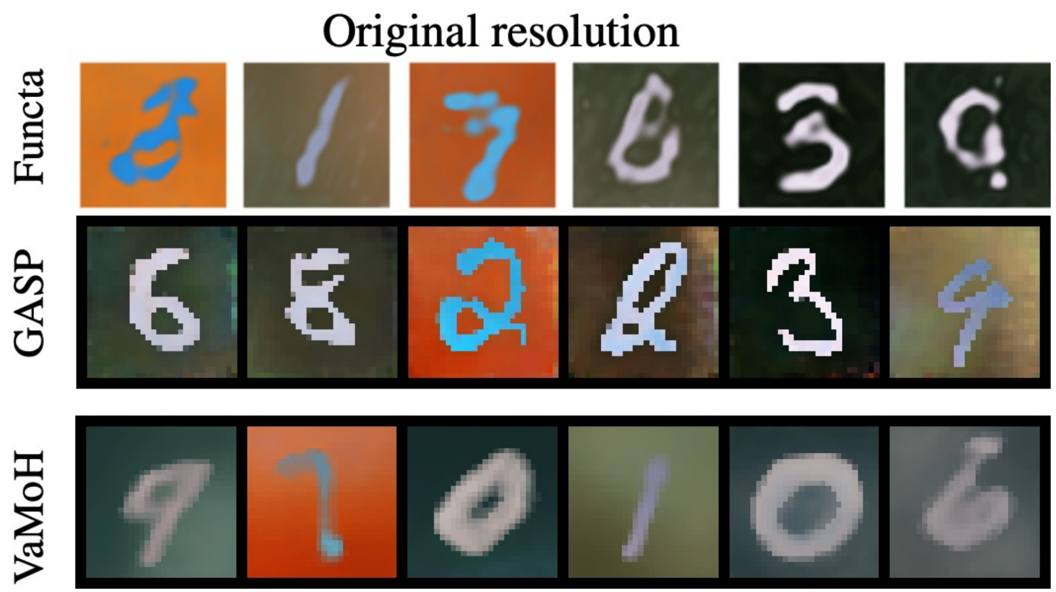


Model	CelebA HQ			Shapes3D		
	$\downarrow$ FID	$\uparrow$ Precision	$\uparrow$ Recall	$\downarrow$ FID	$\uparrow$ Precision	$\uparrow$ Recall
[17] GASP (Dupont et al., 2022b)	**14.01 ± 0.18**	**0.81 ± 0.0**	**0.43 ± 0.01**	118.66 ± 0.64	0.01 ± 0.0	0.16 ± 0.01
[18] Functa (Dupont et al., 2022a)	40.40	-	-	57.81 ± 0.15	0.06 ± 0.0	0.13 ± 0.0
VaMoH	66.27 ± 0.18	0.65 ± 0.0	0.0 ± 0.0	**56.25 ± 0.57**	**0.08 ± 0.0**	**0.64 ± 0.01**

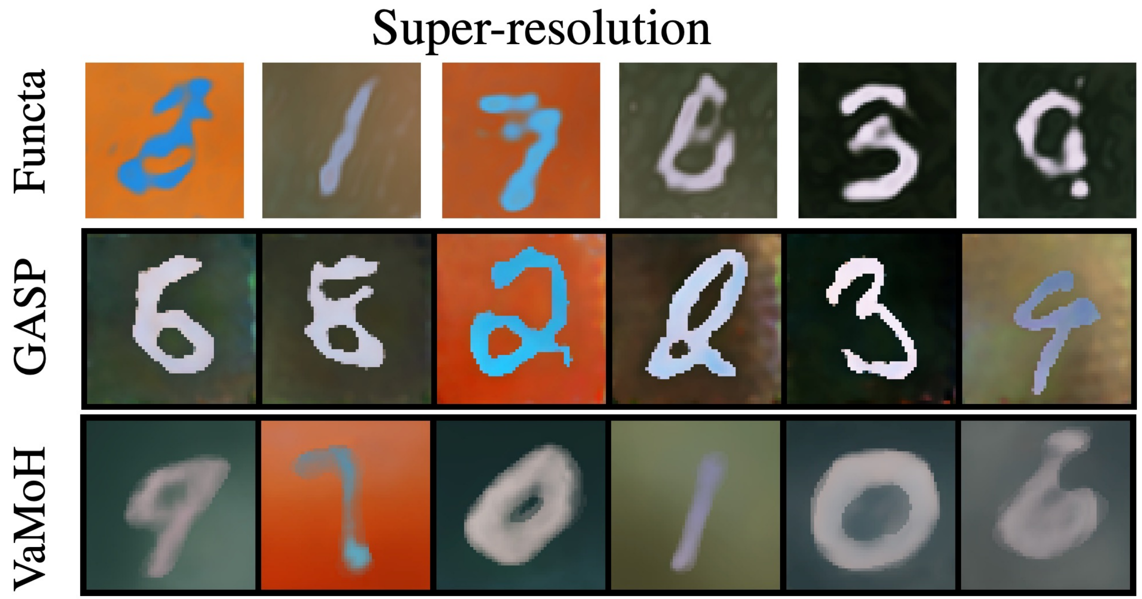
[17] (Dupont et al., 2022a) [18] (Dupont et al., 2022b)

Results

- We naturally generate samples with **any desired resolution**.



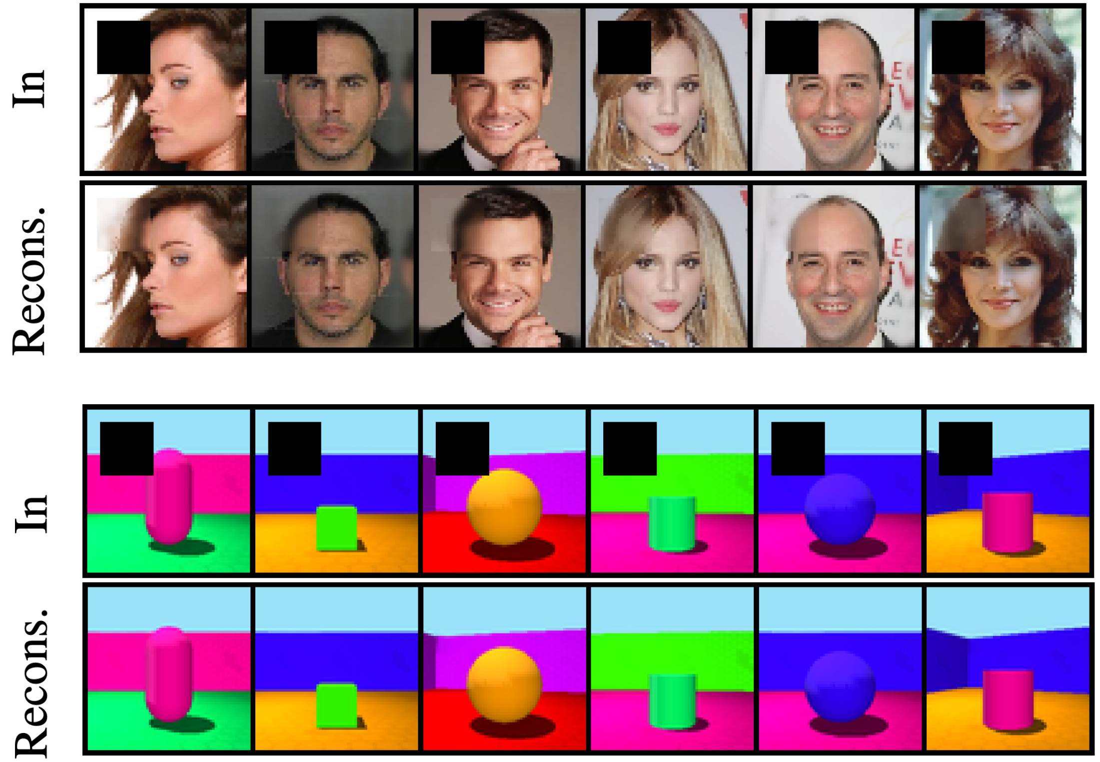
(b) POLYMNIST



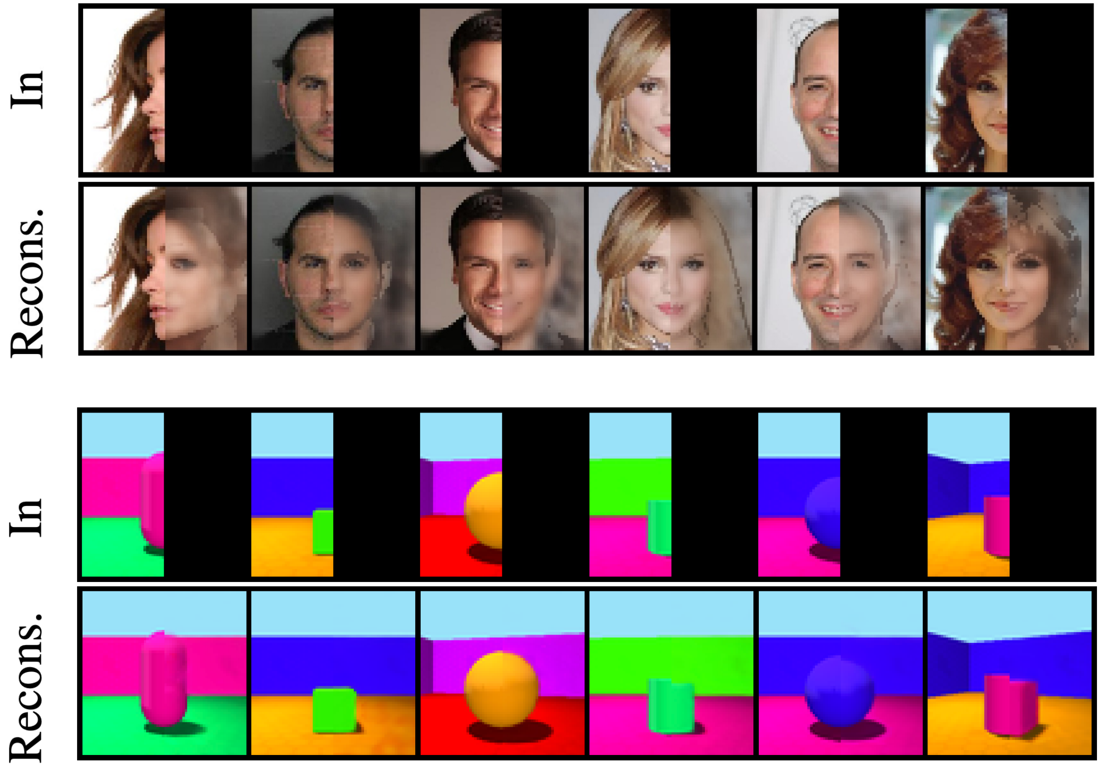
(b) POLYMNIST

Results

- Our method allows for **efficient conditional generation via inference.**



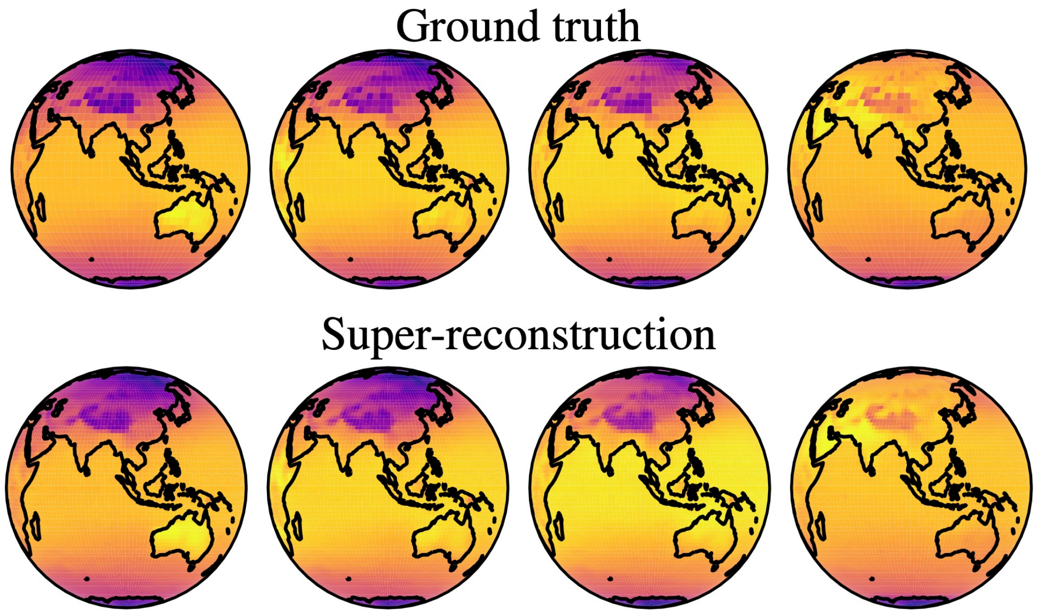
(a) Missing a patch



(b) Missing half of the image

Results

- Our method allows for **efficient conditional generation via inference.**



Results

- We achieve a **7-11 times faster inference than the alternative!**

	Model Inference Time (secs)			Speed Improvement	
Dataset	VaMoH	Functa (3)	Functa (10)	vs. Functa (3)	vs. Functa (10)
POLYMNIST	**0.00453**	0.01648	0.05108	**x 3.64**	**x 11.28**
SHAPES3D	**0.00536**	0.01759	0.05480	**x 3.28**	**x 10.22**
CELEBA HQ	**0.00757**	0.01733	0.05381	**x 2.29**	**x 7.11**
ERA5	**0.00745**	0.01899	0.05932	**x 2.55**	**x 7.96**
SHAPENET	**0.00689**	0.02095	0.06576	**x 3.04**	**x 9.54**

Conclusion

- Thanks to **learning distributions of functions**, our proposed **VAMoH** can:

  - "Sample" neural networks for generating new data.

  - **Infer** the la**tent representation** of a neural network for conditionally generating data.

  - Use the same neural architecture independently of the nature of the data.

  - Easily perform the **conditional generation at any desired resolution**, while being:

    - ✓ **Robust** to partially observed data.
    - ✓ **Expressive** for generating high-quality data.
    - ✓ **Efficient** in terms of inference.

References

[1] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, *27*.

[2] Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

[3] Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. Advances in neural information processing systems, 33, 6840-6851.

[4] Song, Y., & Ermon, S. (2019). Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, *32*.

[5] LeCun, Y., Chopra, S., Hadsell, R., Ranzato, M., & Huang, F. (2006). A tutorial on energy-based learning. *Predicting structured data*, *1*(0).

[6] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[7] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

[8] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

[9] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

[10] Sitzmann, V., Martel, J., Bergman, A., Lindell, D., & Wetzstein, G. (2020). Implicit neural representations with periodic activation functions. *Advances in neural information processing systems*, *33*, 7462-7473.

References

[11] Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., & Geiger, A. (2019). Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4460-4470).

[12] Stanley, K. O. (2007). Compositional pattern producing networks: A novel abstraction of development. Genetic programming and evolvable machines, 8, 131-162.

[13] Ha, D., Dai, A. M., and Le, Q. V. Hypernetworks. In International Conference on Learning Representations, 2017.

[14] Wu, W., Qi, Z., and Fuxin, L. Pointconv: Deep convolutional networks on 3d point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9621–9630, 2019.

[15] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 10684-10695).

[16] Rezende, D. and Mohamed, S. Variational inference with normalizing flows. In International Conference on Machine Learning, 2015.

[17] Dupont, E., Teh, Y. W., and Doucet, A. Generative models as distributions of functions. In International Conference on Artificial Intelligence and Statistics, pp. 2989–3015. PMLR, 2022a.

[18] Dupont, E., Kim, H., Eslami, S. A., Rezende, D. J., and Rosenbaum, D. From data to functa: Your data point is a function and you can treat it like one. In International Conference on Machine Learning, pp. 5694–5725. PMLR, 2022

Andaluz.IA

Universidad Carlos III de Madrid

Thank you!

[Paper]

[Code]