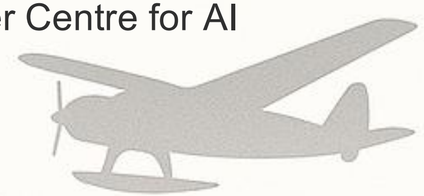
HYPER-TRANSFORMING LATENT DIFFUSION MODELS

Ignacio Peis
Technical University of Denmark
Pioneer Centre for AI
ipeaz@dtu.dk

Batuhan Koyuncu
Saarland University

Isabel Valera
Saarland University

Jes Frellsen
Technical University of Denmark
Pioneer Centre for AI



DTU

UNIVERSITÄT DES SAARLANDES

PIONEER CENTRE FOR ARTIFICIAL INTELLIGENCE

ICML International Conference On Machine Learning

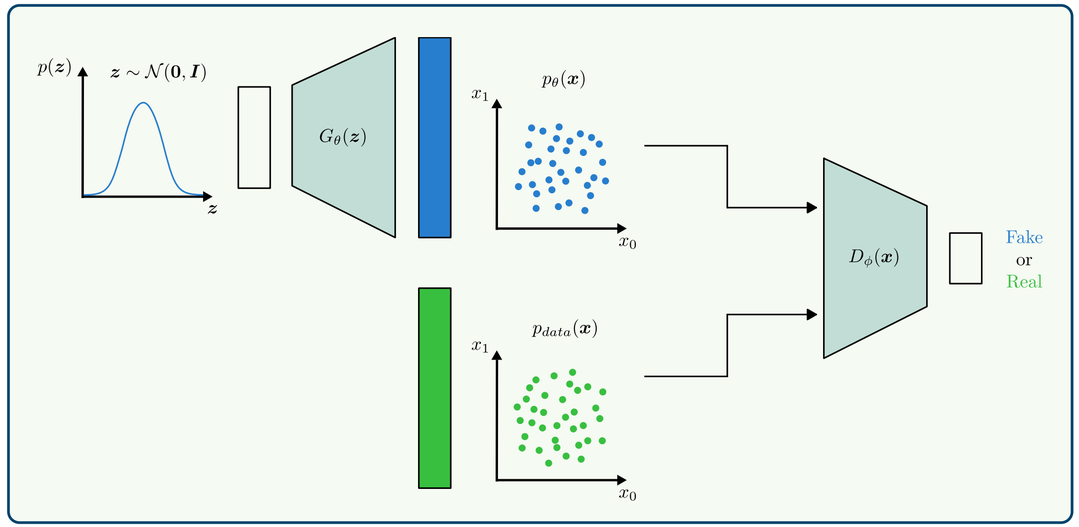
Andaluz.IA

Introduction

Deep Generative Models

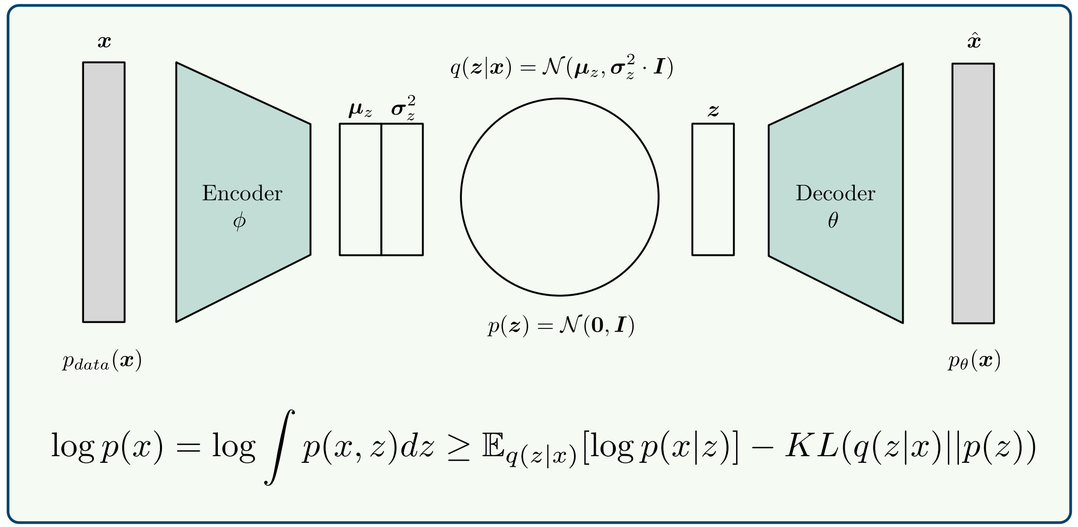
- Learning **probability distributions** on data using Deep Neural Networks.

Implicitly



Generative Adversarial Networks (GANs)

Explicitly



Variational Autoencoders (VAEs)
Diffusion Models
Flow-based models
Energy-based models

[0] Peis, 2023

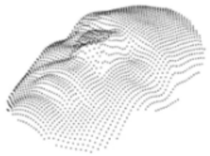
Introduction

Discretization of signals

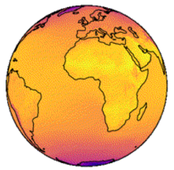
- We typically deal with discretized versions of data that are continuous in nature.

2D Images

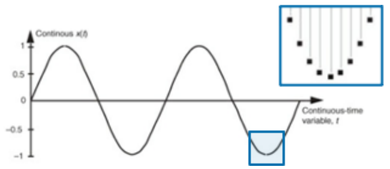
3D Images



Polar data

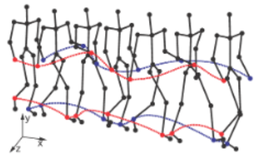


Time series

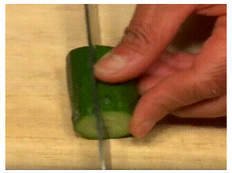


Audio

Motion sequences



Video



Spatial

Temporal

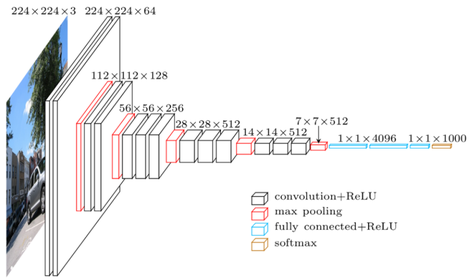
Spatio-temporal

Introduction

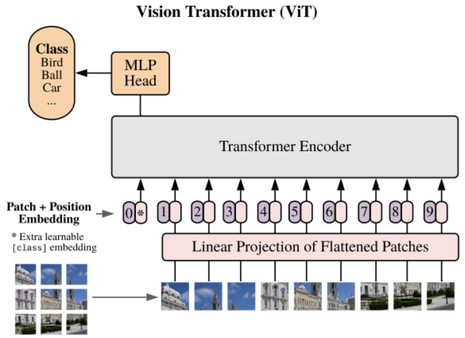
NNs to exploit discretized data

CNNs, Vision Transformers

[6]



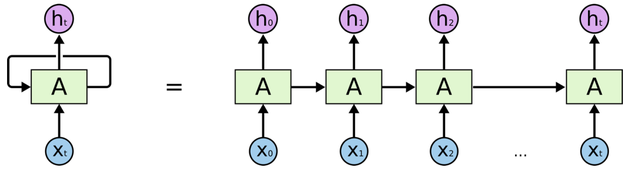
[7]



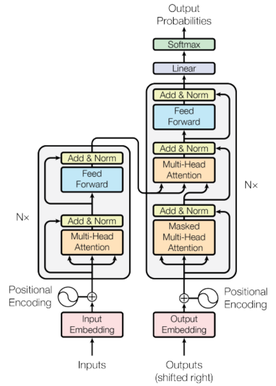
Spatial

RNNs, Transformers

[8]

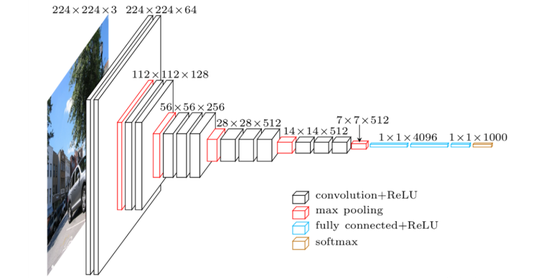


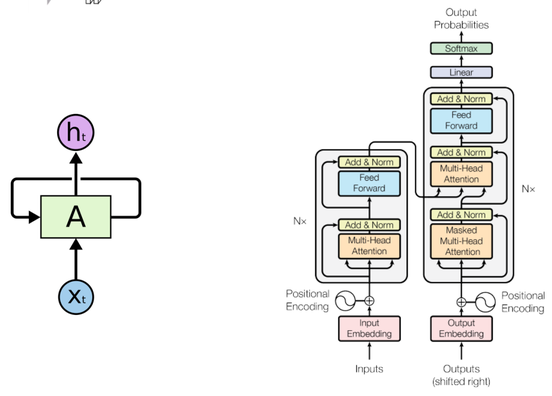
[9]



Temporal

CNNs, RNNs, Transformers





Spatio-temporal

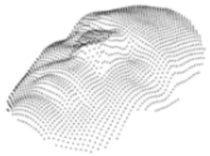
Introduction

Real data is continuous in nature

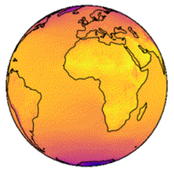
- Every discretized observation comes from an underlying continuous signal.

2D Images

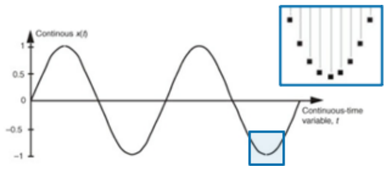
3D Images



Polar data

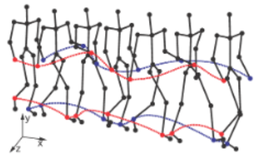


Time series

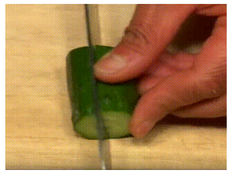


Audio

Motion sequences



Video



Spatial

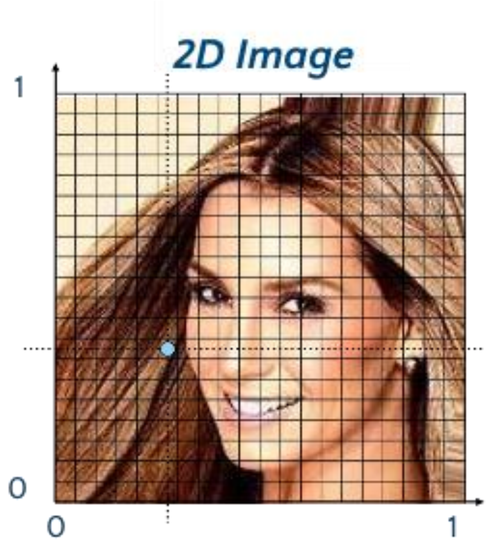
Temporal

Spatio-temporal

$f : \mathbb{R}^2 \to \mathbb{R}^3, f(x_1, x_2) = (r, g, b)$ $f : \mathbb{R}^3 \to \{0, 1\}, f(x_1, x_2, x_3) = p$ $f : \mathbb{R}^2 \to \mathbb{R}, f(\varphi, \lambda) = T$ $f : \mathbb{R}^3 \to \mathbb{R}^3, f(x_1, x_2, t) = (r, g, b)$

Introduction

- Focusing on images:

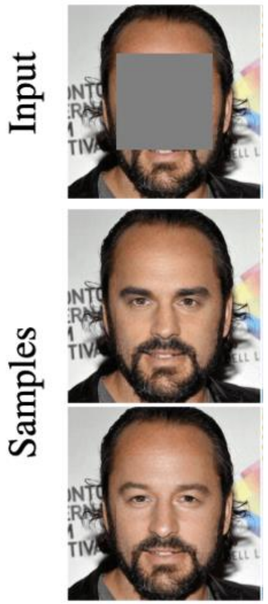


- Generator function $f : \boldsymbol{X} \rightarrow \boldsymbol{Y}$ creates this speficic image with the mapping $f(\boldsymbol{x}_d) = \boldsymbol{y}_d$, $d \in [1, ..., D]$
- Each pixel is now a pair $\{\boldsymbol{x}_d, \boldsymbol{y}_d\}$ where $x_d \in \mathbb{R}^2$, $y_d \in \mathbb{R}^3$
- Full image is a pair of sets $\boldsymbol{X} = \{\boldsymbol{x}_d\}_{d=1}^{D}, \quad \boldsymbol{Y}_d = \{\boldsymbol{y}_d\}_{d=1}^{D}$

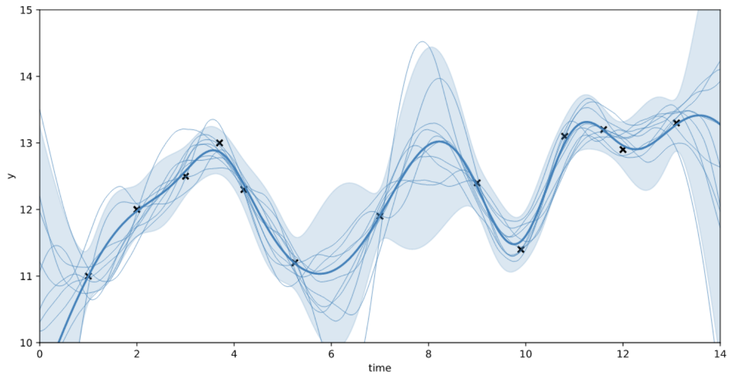
Introduction

- Learning to generative in the space of functions allows for naturally handling:

Inpainting



Conditional generation

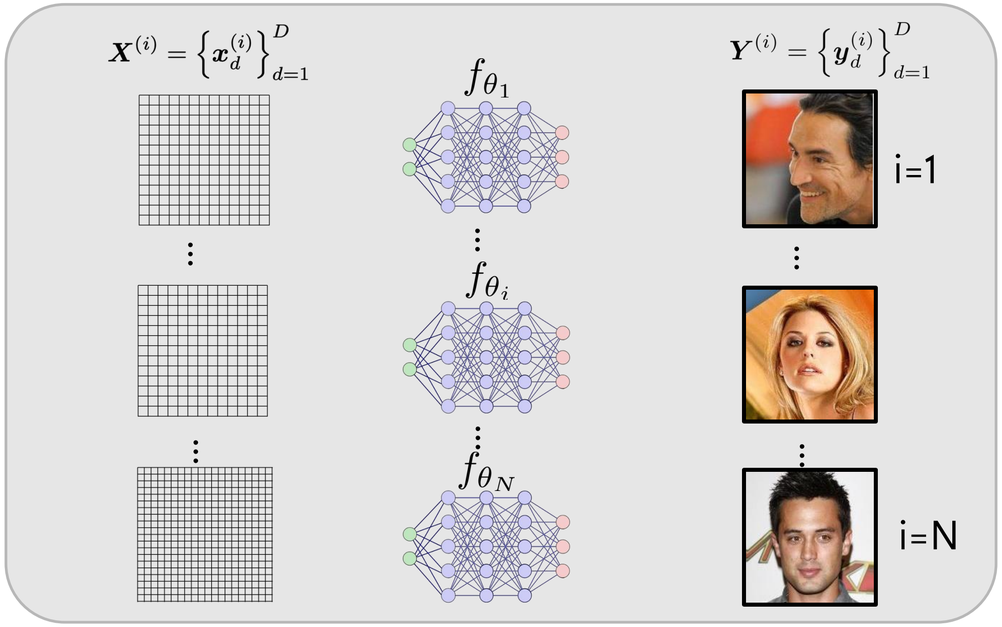


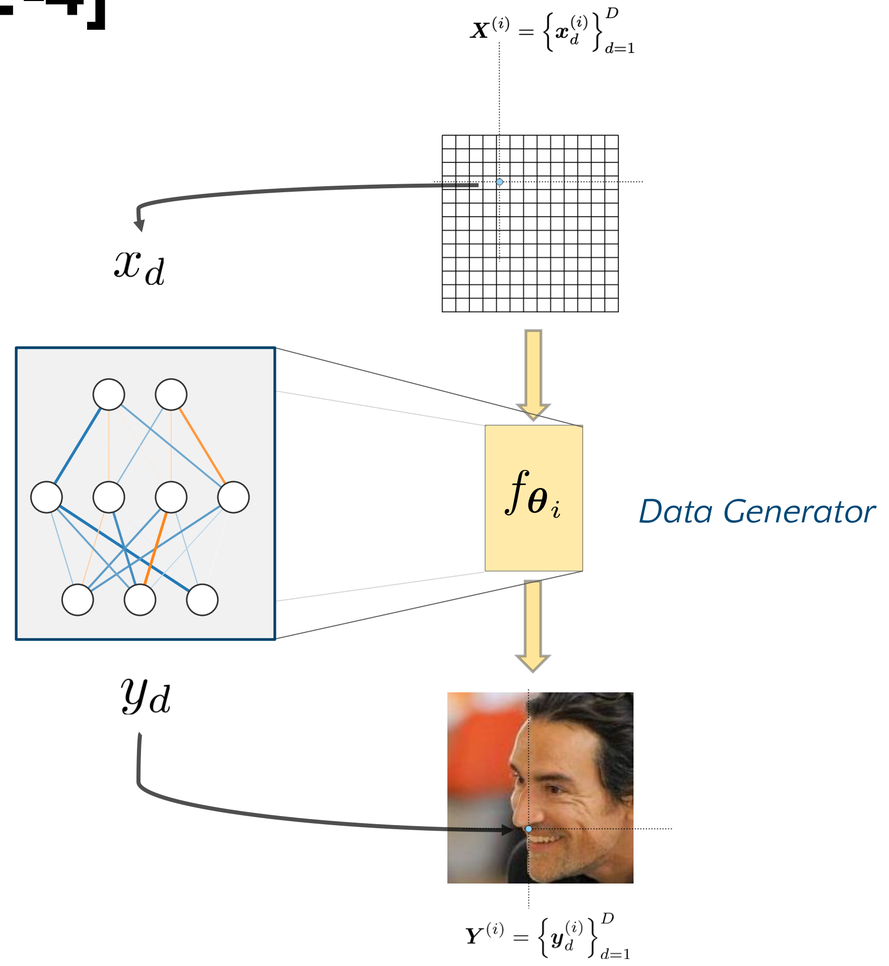
Super-resolution

Introduction

Implicit Neural Representations (INRs) [2-4]

Data generator $f_{\boldsymbol{\theta}_i}$ is unique to each image





[2] Sitzmann et at., 2020 [3] Mescheder et at., 2019 [4] Sitzmann et at., 2019

Introduction

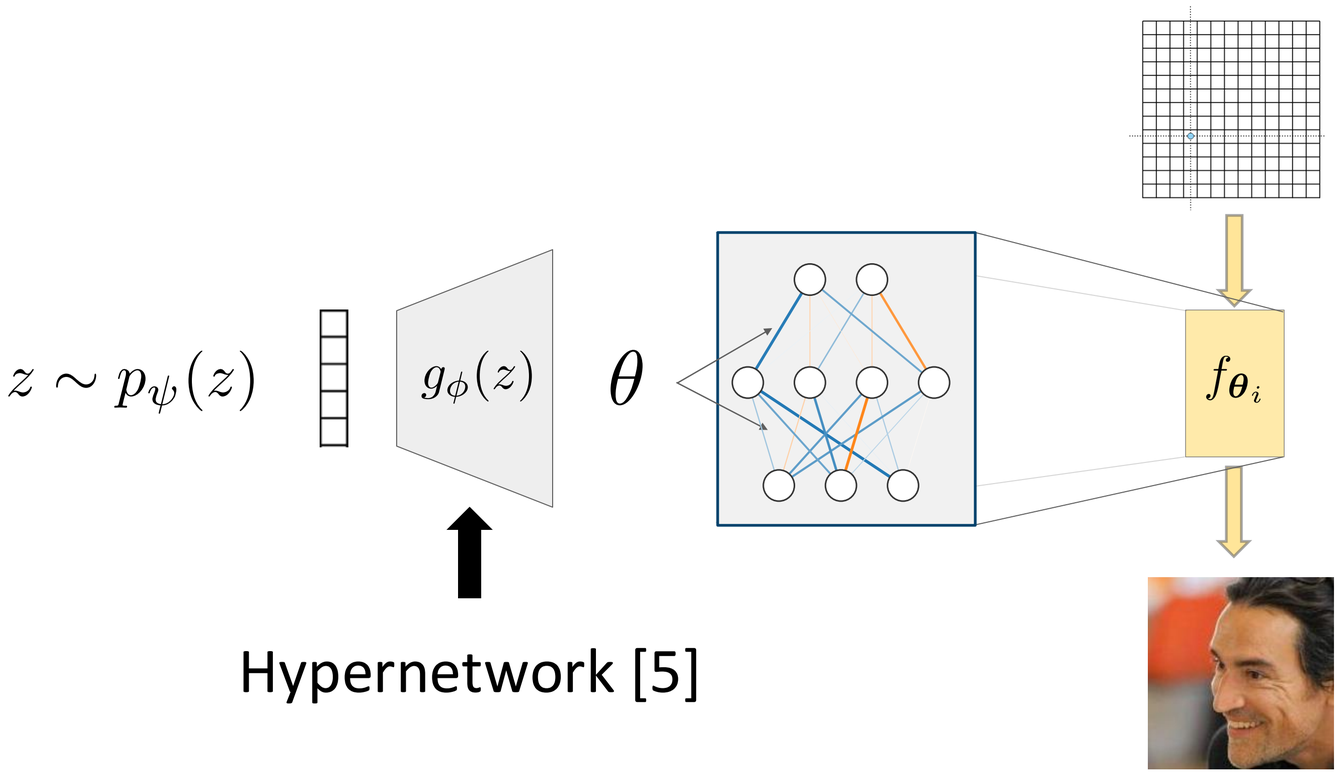
How to scale to large datasets?

How to map data to an INR?

Introduction

Hypernetworks [5]

Have $\boldsymbol{z}^{(i)}$, a summary representation of image.



[5] Ha et at., 2017

Introduction

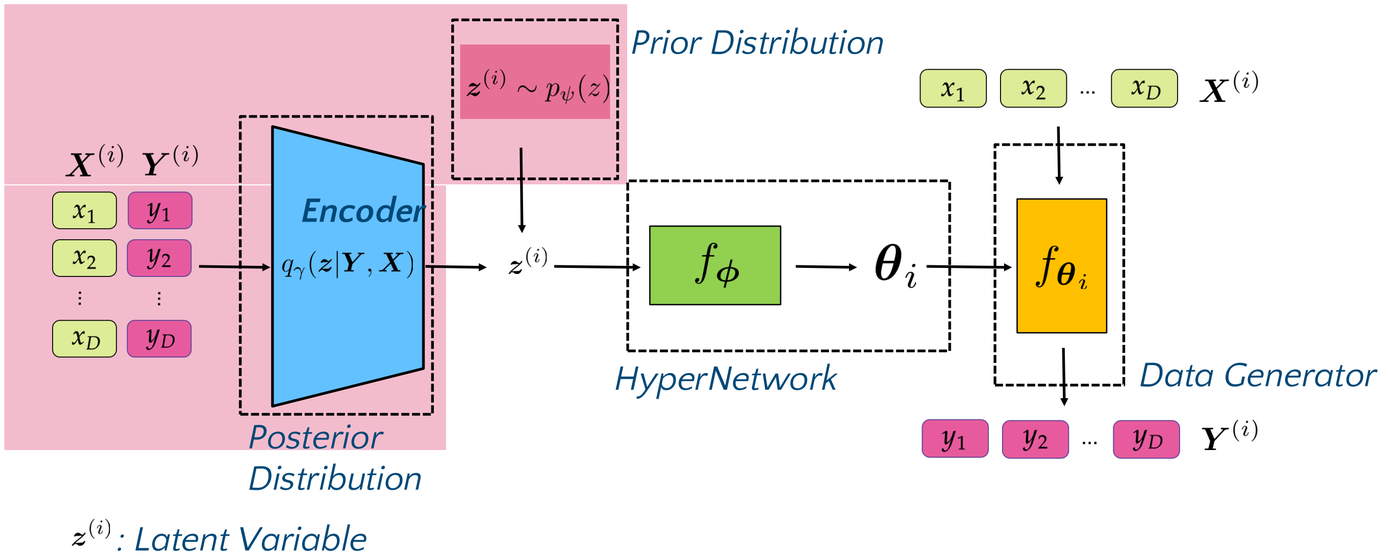
How to infer the latent representation ***z***?

$$p_\theta(\boldsymbol{z})$$

Related Work

VAMoH [9]

- Variational Inference
  - Requires flexible, learnable prior.



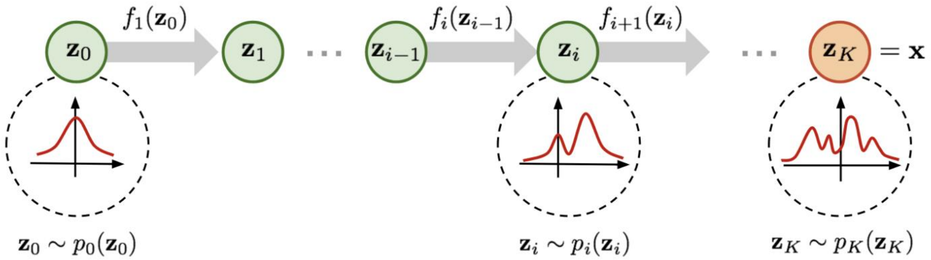
$$\mathcal{L}(\boldsymbol{Y}, \boldsymbol{X}; \psi, \phi, \boldsymbol{\gamma}) = \sum_{d=1}^{D} \mathbb{E}_{q_{\gamma_z}(\boldsymbol{z}|\boldsymbol{Y},\boldsymbol{X})} \left[\sum_{k=1}^{K} \log p_{\boldsymbol{\theta}_k} (\boldsymbol{y}_d \mid \boldsymbol{x}_d) \cdot \pi_{dk} \right] - D_{KL} \left(q_{\gamma_z}(\boldsymbol{z}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

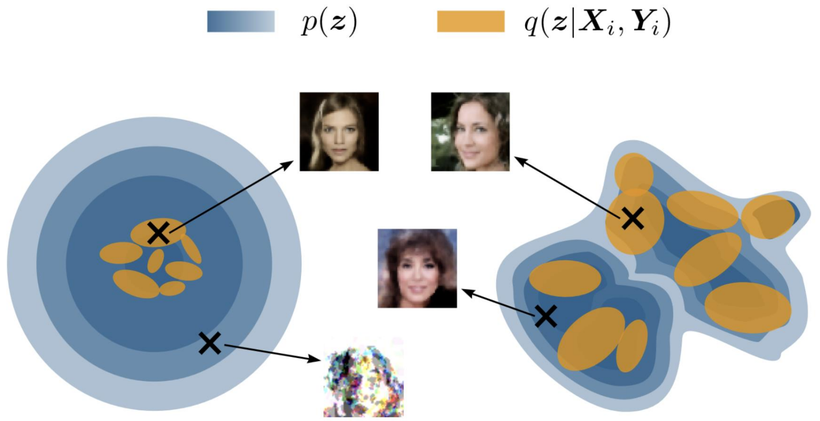
[9] Koyuncu et at., 2023

Related Work

VAMoH [9]

- Variational Inference
  - Requires flexible, learnable prior.





The ***hole*** problem

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

[9] Koyuncu et at., 2023

Motivation

Flexibility of the latent space in [6, 7, 9]

✕ Poor generation quality.

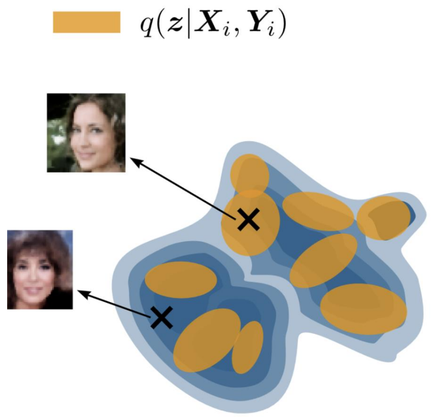
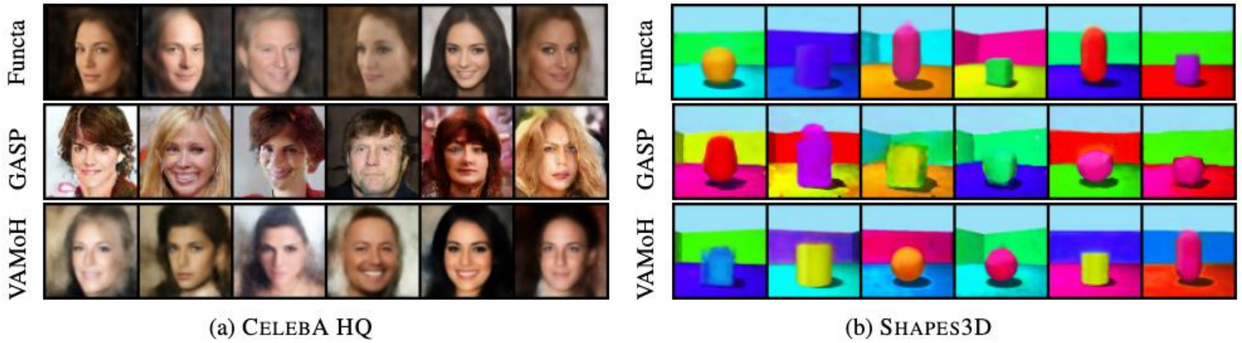


Image Reconstruction with VAMoH

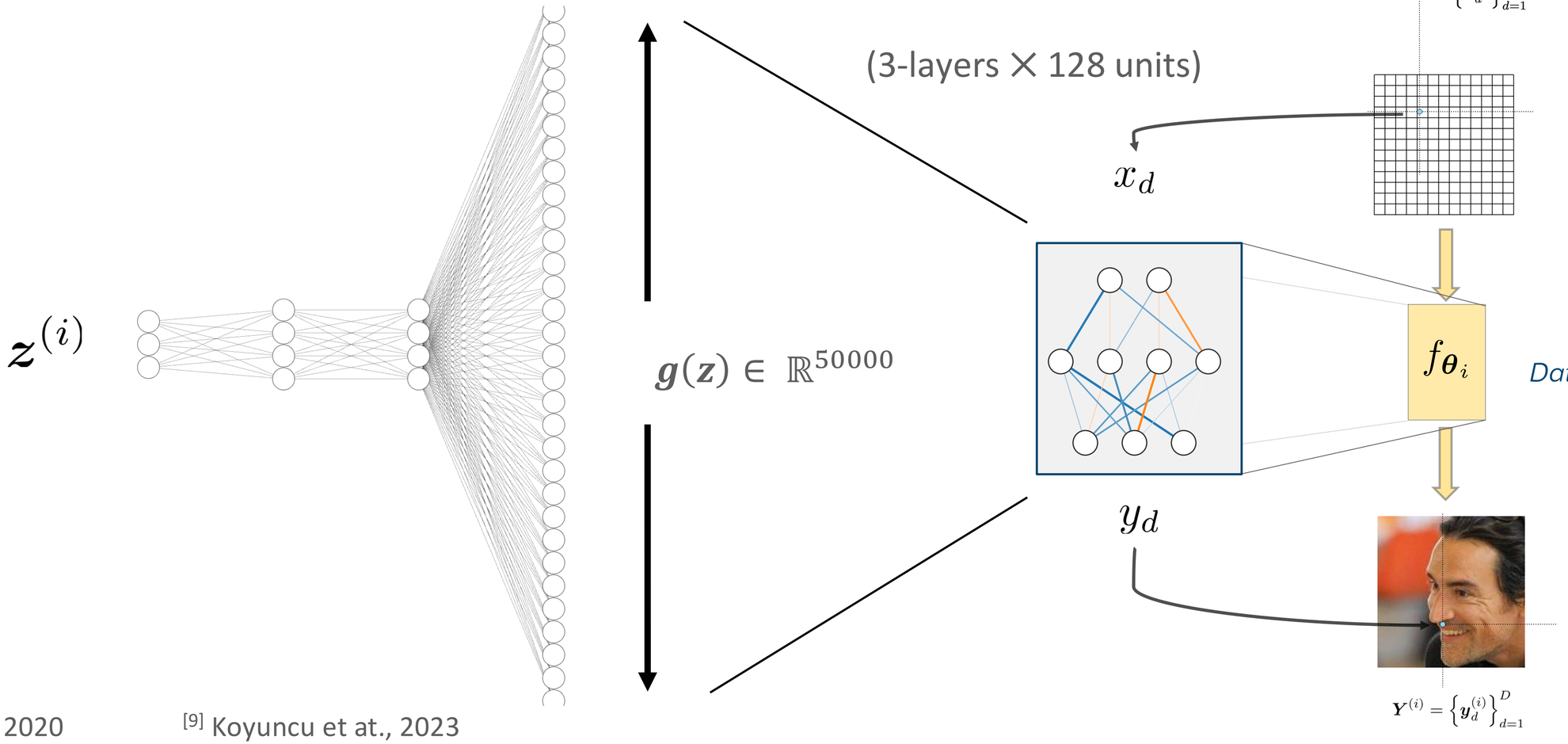


(a) CelebA HQ (b) Shapes3D

[6] Dupont et at., 2020 [7] Dupont et at., 2022 [9] Koyuncu et at., 2023

Motivation

Hypernetwork bottleneck in [6, 9]



[6] Dupont et at., 2020

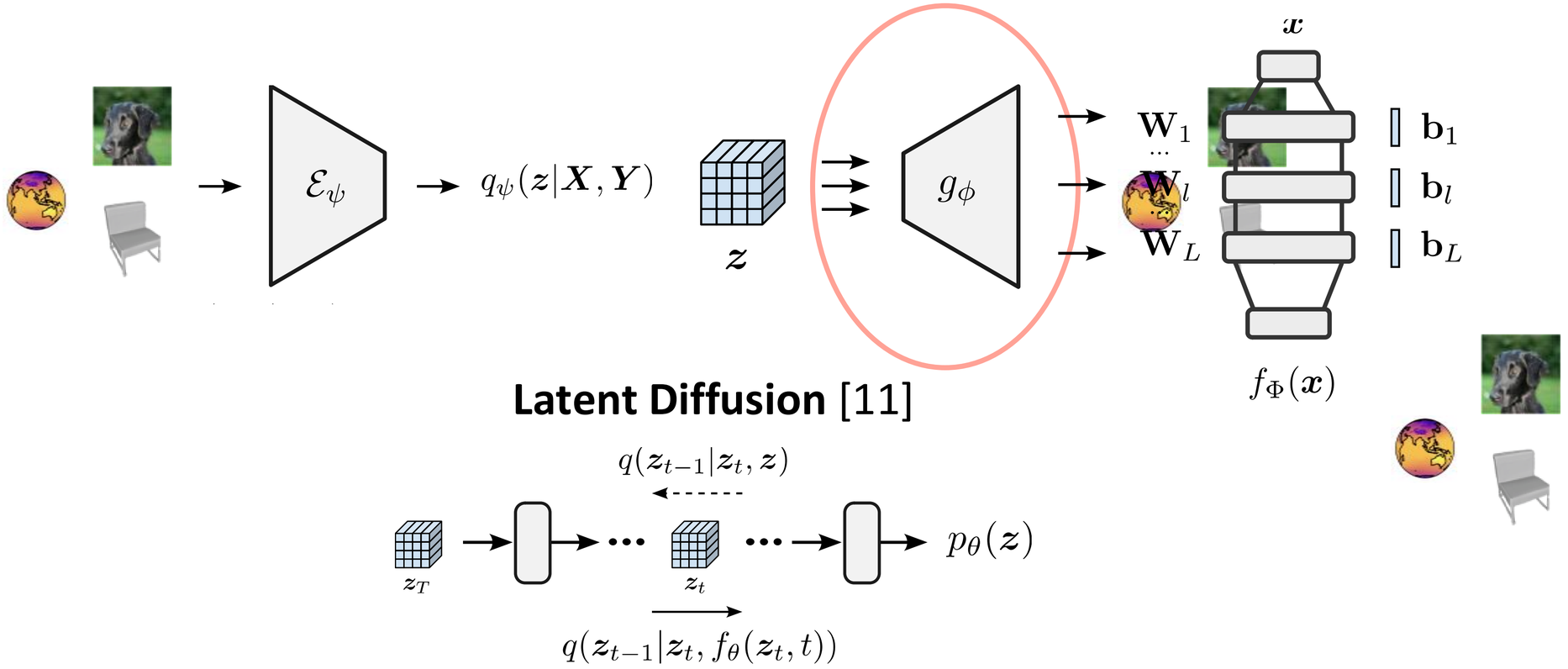
[9] Koyuncu et at., 2023

Proposed method

Latent Diffusion Models of INRs

LDMI [1]

Transformer-based Hypernetwork [10]

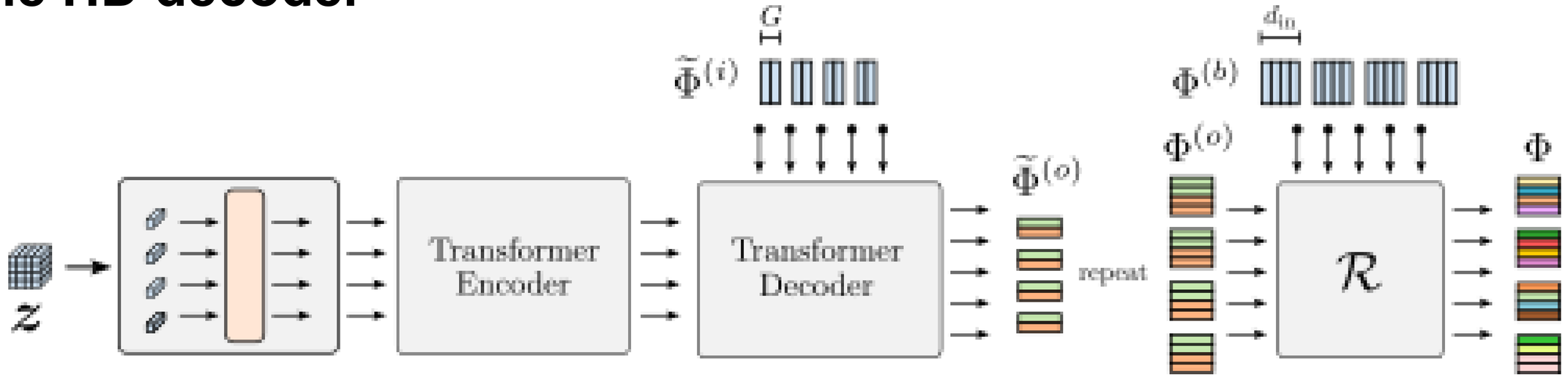


Latent Diffusion [11]

[1] Peis et at., 2025 [10] Chen et at., 2024 [11] Rombach et at., 2021

Latent Diffusion Models of INRs

The HD decoder



- The **latent tokens** come from tokenizing ***z*** (following ViT [32]) and processing them via a Transformer Encoder.
- The **weight tokens** are columns of the weight matrices.
- To trade-off representation capacity vs computational feasibility, we learn a subset of the columns, *G*.

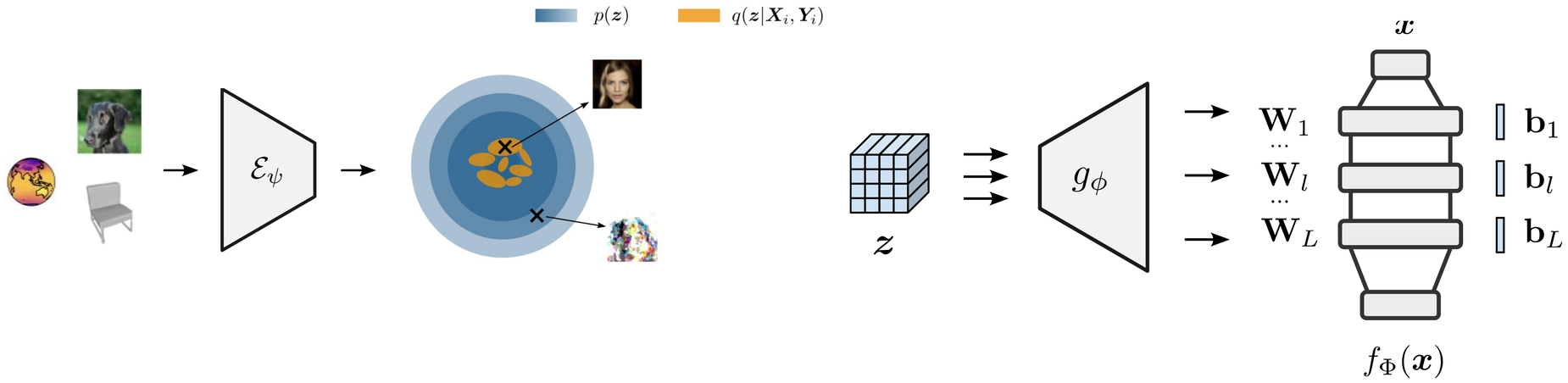
Latent Diffusion Models of INRs

The HD decoder

We will **firstly** train an *"under-regularized"* autoencoder to accurately represent data in a (tensor-shaped) latent space.

- The latents are mapped into INRs using our **transformer–based hypernetwork decoder**.

$$\mathcal{L}_{\text{VAE}}(\phi, \psi) = \mathbb{E}_{q_\psi(\boldsymbol{z}|\boldsymbol{X})} \left[\log p_\Phi(\boldsymbol{X})\right] - \beta \cdot D_{\text{KL}} \left(q_\psi(\boldsymbol{z} \mid \boldsymbol{X}) \| p(\boldsymbol{z})\right) + \mathcal{L}_{\text{perceptual}} + \mathcal{L}_{\text{GAN}}$$

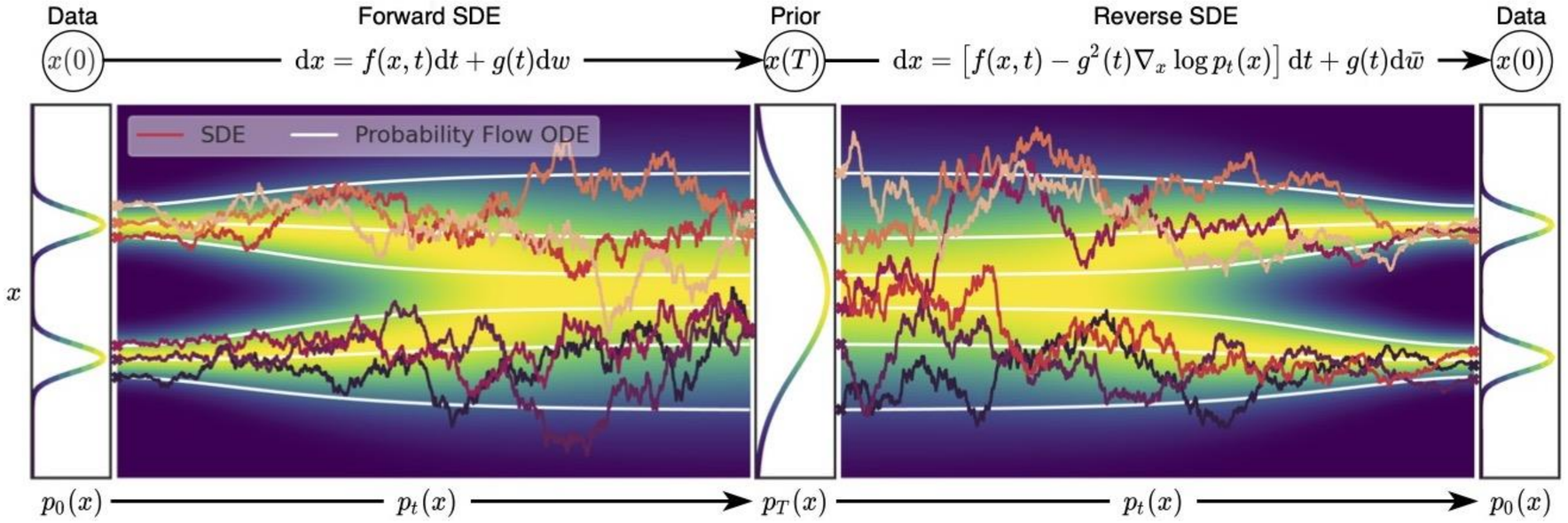


Latent Diffusion Models of INRs

Diffusion Models [12]

Denoising Score Matching

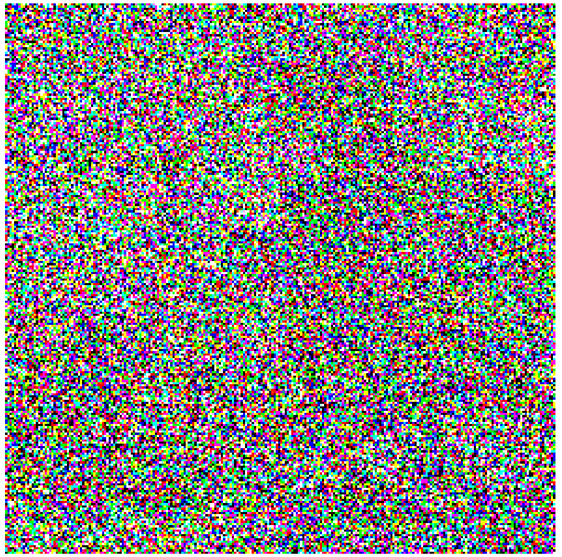
$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathbb{E}_t \left\{ \lambda(t) \mathbb{E}_{\mathbf{x}(0)} \mathbb{E}_{\mathbf{x}(t)|\mathbf{x}(0)} \left[\left\| \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}(t), t) - \nabla_{\mathbf{x}(t)} \log p_{0t}(\mathbf{x}(t) \mid \mathbf{x}(0)) \right\|_2^2 \right] \right\}$$

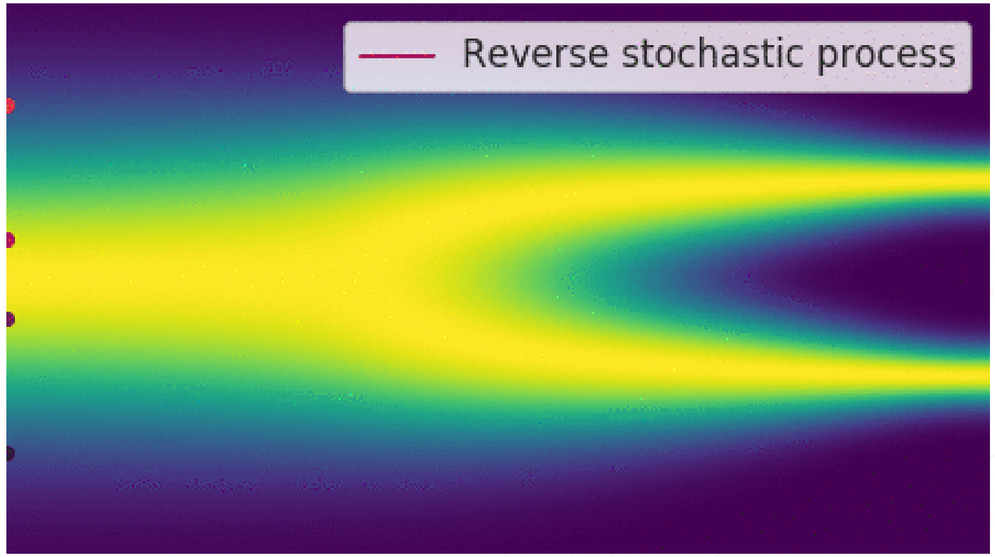


[12] Song et at., 2020

Latent Diffusion Models of INRs

Diffusion Models [12]

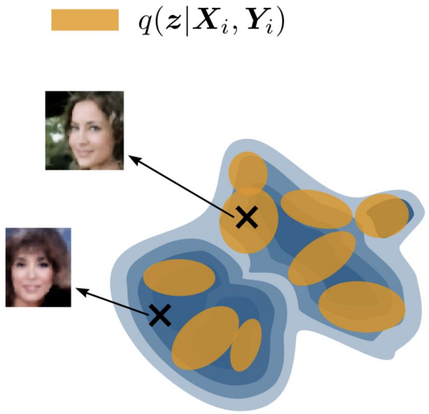




[12] Song et at., 2020

Latent Diffusion Models of INRs

Latent Diffusion Models [11]

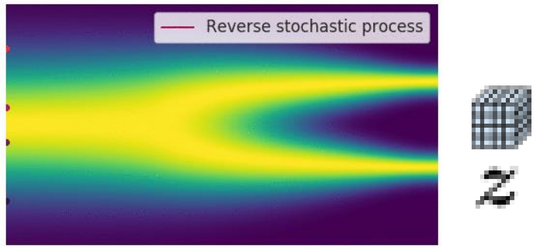


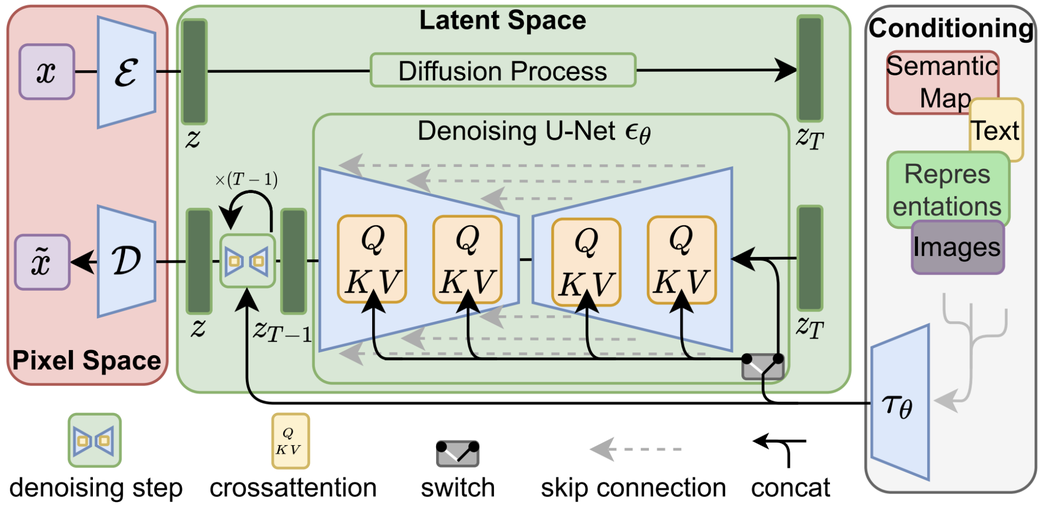
- First stage:

$$\mathcal{L}_{\text{VAE}}(\phi, \psi) = \mathbb{E}_{q_\psi(\boldsymbol{z}|\boldsymbol{X})} \left[\log p_\Phi(\boldsymbol{X})\right] - \beta \cdot D_{\text{KL}} \left(q_\psi(\boldsymbol{z} \mid \boldsymbol{X}) \| p(\boldsymbol{z})\right), + \mathcal{L}_{\text{perceptual}} + \mathcal{L}_{\text{GAN}}$$

- Second stage:

$$\mathcal{L}_{\text{DDPM}} = \mathbb{E}_{\boldsymbol{X}, \boldsymbol{z}, \epsilon, t} \left[\lambda(t) \left\| \epsilon - \epsilon_\theta \left(\boldsymbol{z}_t, t\right) \right\|^2\right],$$





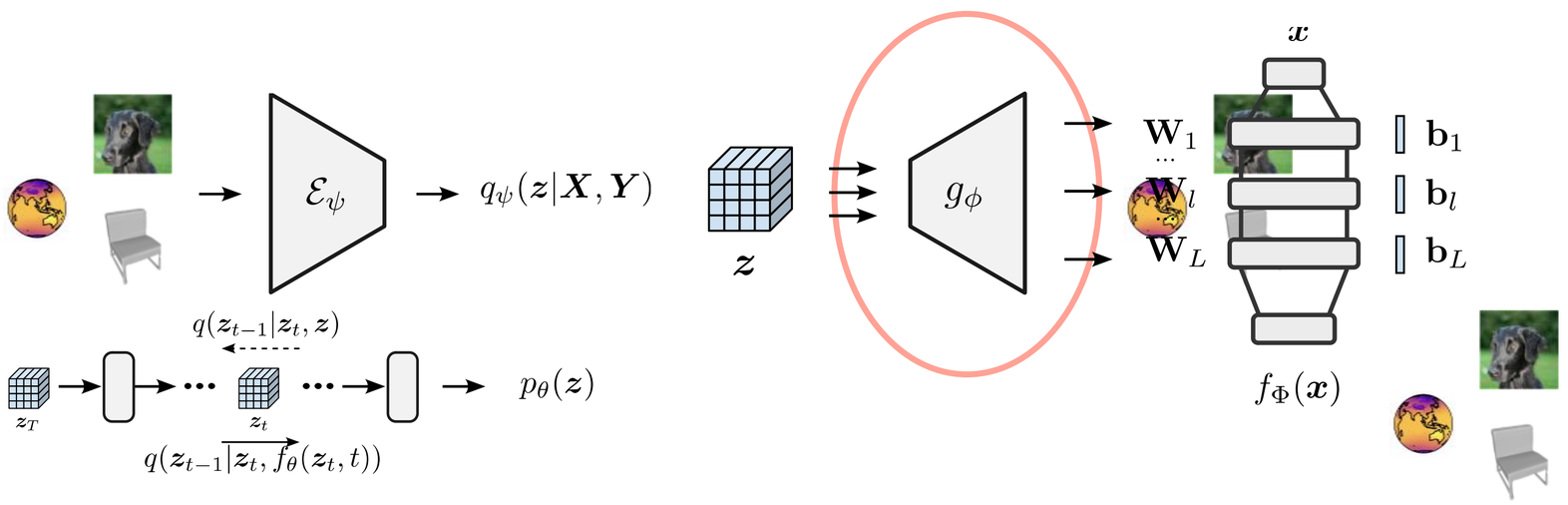
[11] Rombach et at., 2021

Latent Diffusion Models of INRs

Hyper-Transforming

- We can download pre-trained LDMs and just re-train only our decoder!

$$\mathcal{L}_{\text{HT}}(\phi) = \mathbb{E}_{q_\psi(\boldsymbol{z}|\boldsymbol{X},\boldsymbol{Y})}\left[\log p_\Phi(\boldsymbol{Y} \mid \boldsymbol{X})\right] + \mathcal{L}_{\text{perceptual}} + \mathcal{L}_{\text{GAN}}$$



Pretrained LDMs

Datset	Task	Model	FID	IS	Prec	Recall	
CelebA-HQ	Unconditional Image Synthesis	LDM-VQ-4 (200 DDIM steps, eta=0)	5.11 (5.11)	3.29	0.72	0.49	https://omr diffusion/c
FFHQ	Unconditional Image Synthesis	LDM-VQ-4 (200 DDIM steps, eta=1)	4.98 (4.98)	4.50 (4.50)	0.73	0.50	https://omr diffusion/ff
LSUN-Churches	Unconditional Image Synthesis	LDM-KL-8 (400 DDIM steps, eta=0)	4.02 (4.02)	2.72	0.64	0.52	https://omr diffusion/ls

Experiments

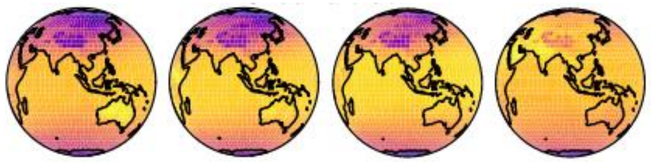
Datasets

CelebA-HQ (64x64)

CelebA-HQ (256x256)

ImageNet (256x256)

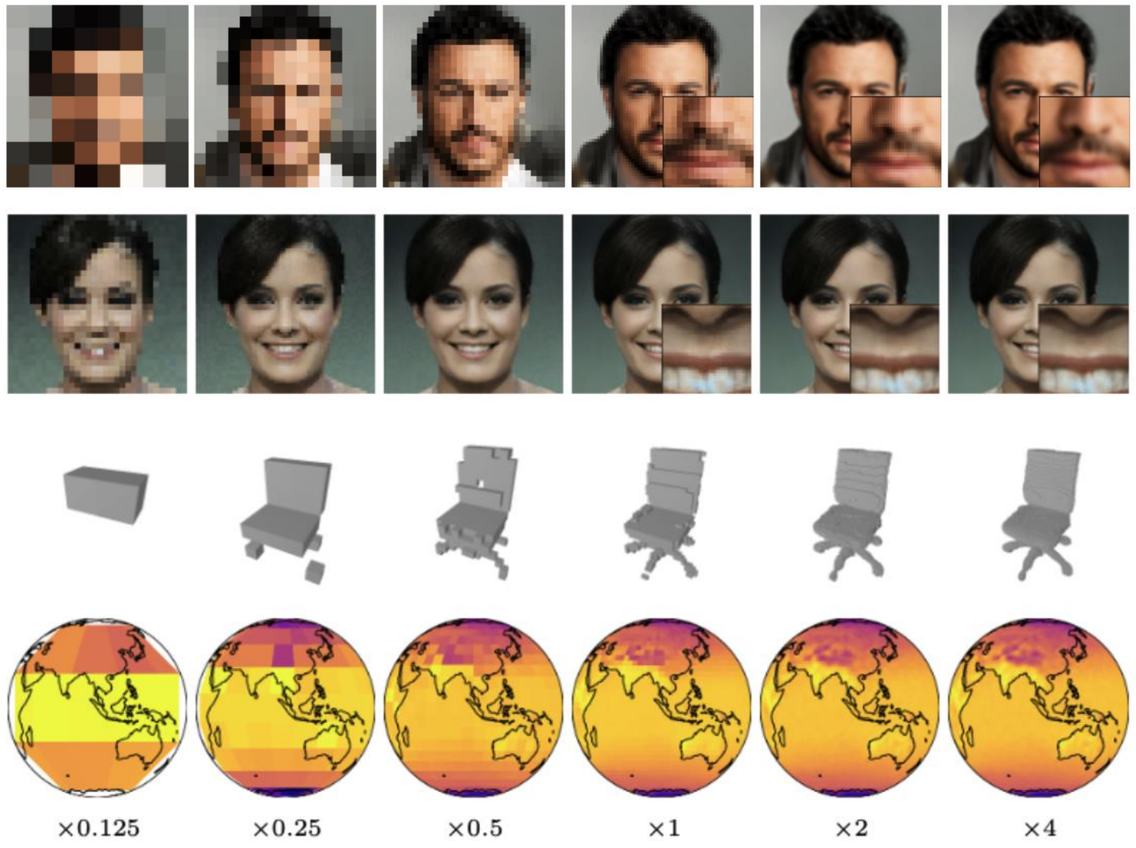
ERA5 (Polar)

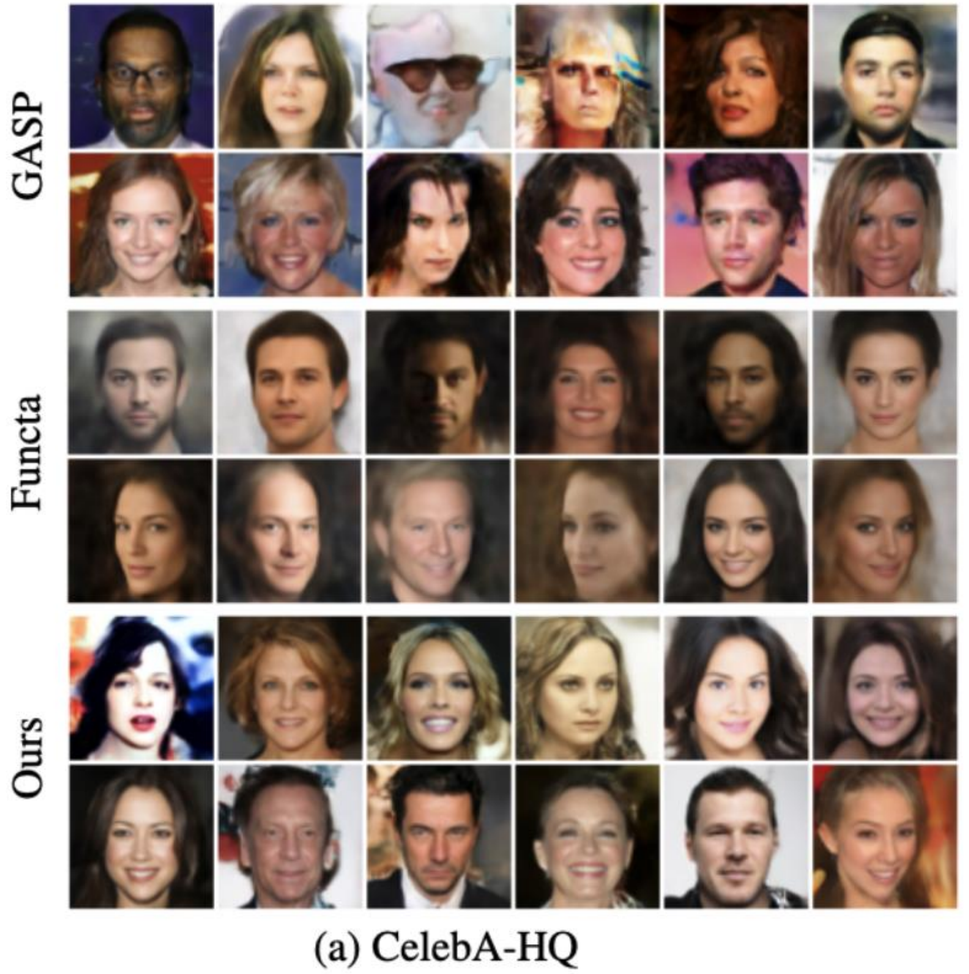


ShapeNET (Voxels)

Experiments

Generation





(a) CelebA-HQ

Experiments

Reconstruction

Model	PSNR (dB) ↑	FID ↓	HN Params ↓
CelebA-HQ (64 × 64)			
GASP [Dupont et al., 2022a]	-	**7.42**	25.7M
Functa [Dupont et al., 2022b]	≤ **30.7**	40.40	-
VAMoH [Koyuncu et al., 2023]	23.17	66.27	25.7M
LDMI	24.80	18.06	**8.06M**
ImageNet (256 × 256)			
Spatial Functa [Bauer et al., 2023]	≤ 38.4	≤ 8.5	-
LDMI	20.69	**6.94**	102.78M

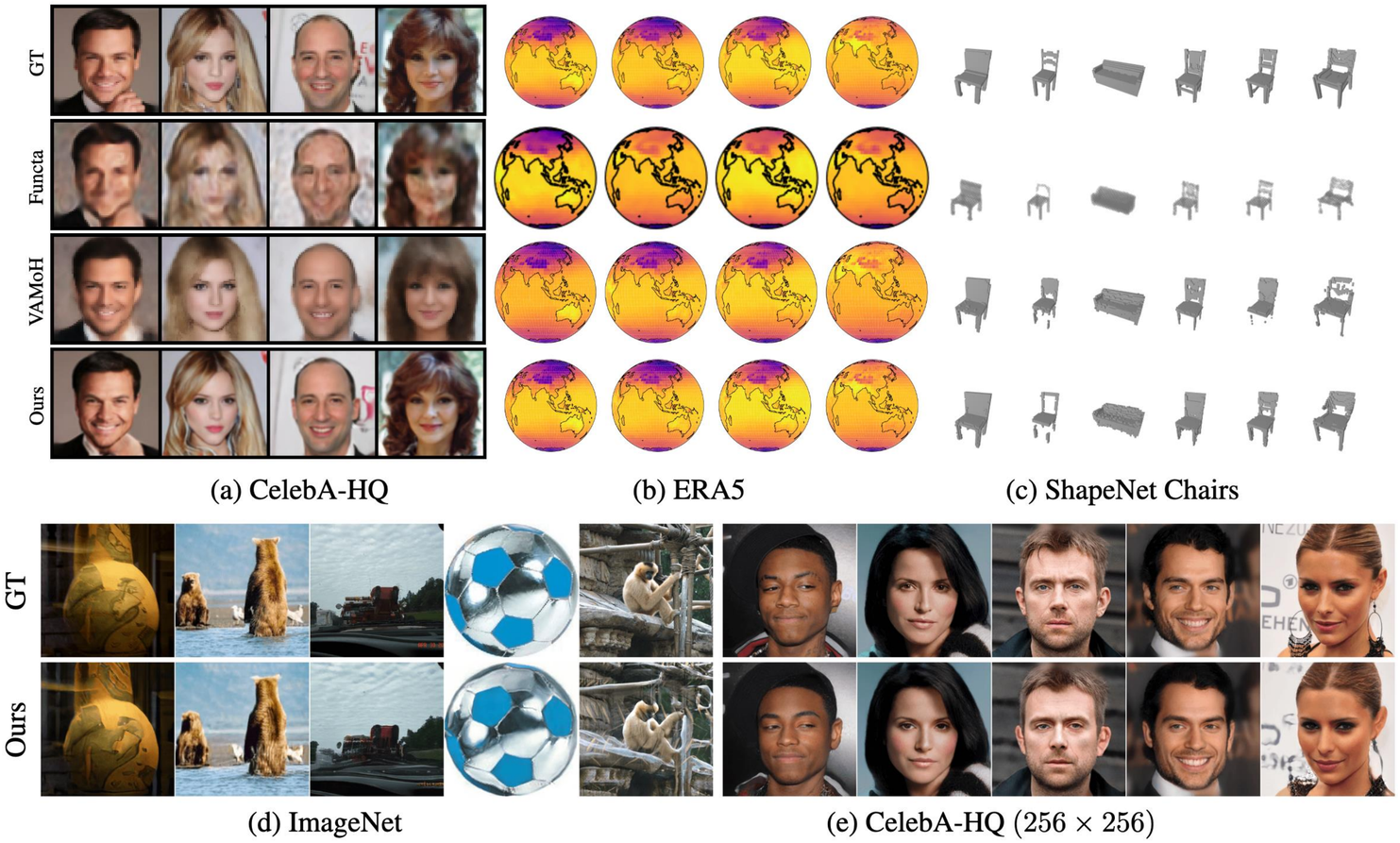
Table 1: Metrics on CelebA-HQ and ImageNet.

Model	Chairs (PSNR) ↑	ERA5 (PSNR) ↑
Functa [Dupont et al., 2022b]	29.2	34.9
VAMoH [Koyuncu et al., 2023]	38.4	39.0
LDMI	**38.8**	**44.6**

Table 2: Reconstruction quality (PSNR in dB) on ShapeNet Chairs and ERA5 climate data, demonstrating LDMI's strong generalization capabilities across modalities. Note that GASP is omitted as it is not applicable to INR reconstruction tasks.

Method	HN Params	INR Weights	Ratio (INR/HN)
GASP/VAMoH	25.7M	50K	0.0019
LDMI	**8.06M**	**330K**	**0.0409**

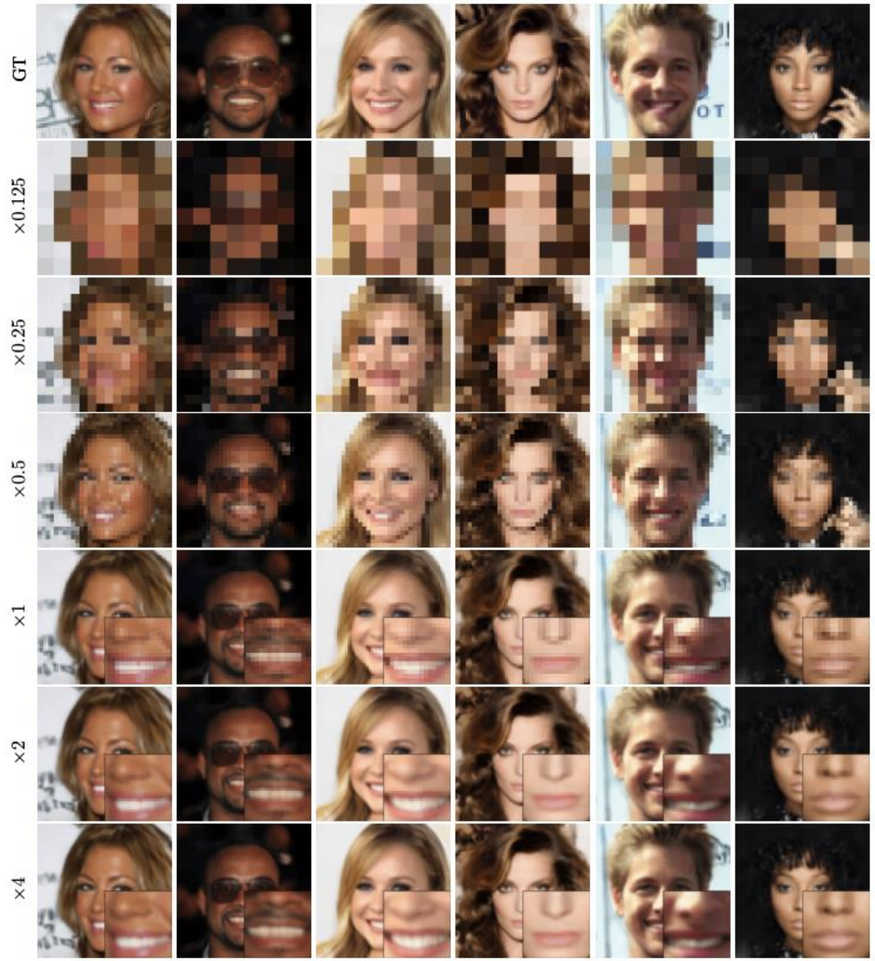
Table 3: Parameter efficiency of LDMI.



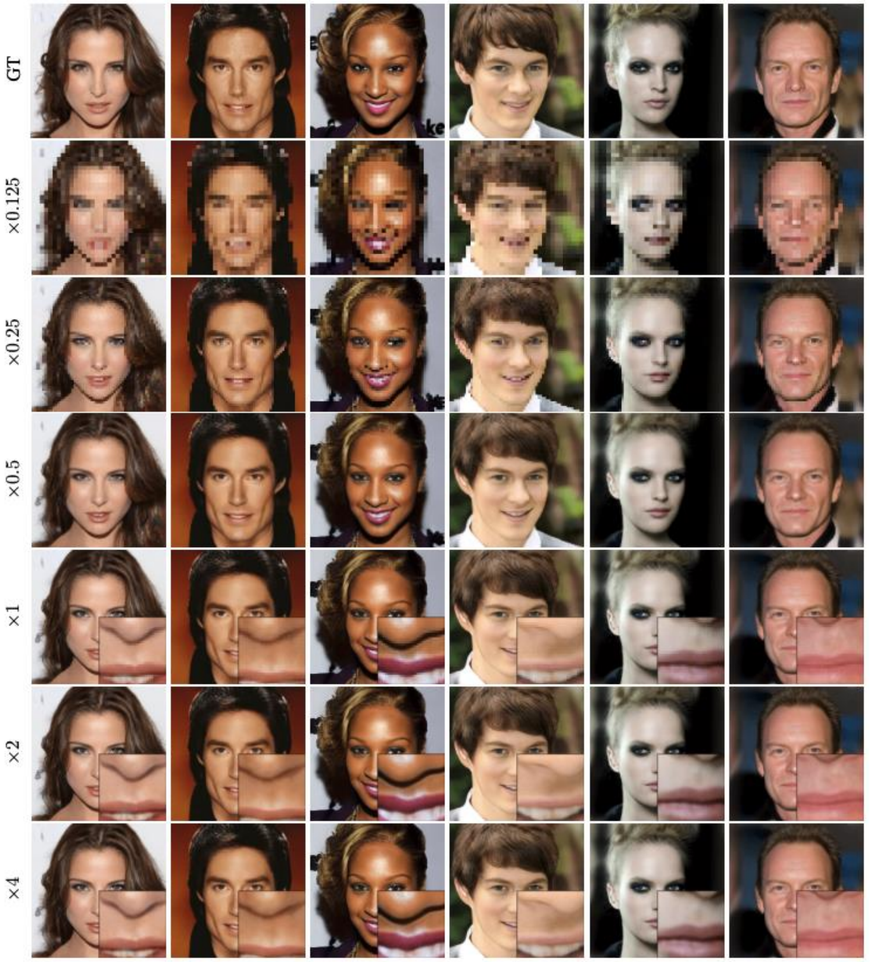
Experiments

Reconstruction

CelebA-HQ (64x64)



CelebA-HQ (256x256)



Experiments

Data completion

VAMoH

LDMI

Conclusion

Thanks to using **Latent Diffusion** and a novel **Transformer-based hypernetwork**, **LDMI** enhances

- Resolution-agnostic generation.
- Resolution-agnostic reconstruction.

While:

✓ Being scalable.

✓ Being efficient in parameter usage.

✓ Working with multiple data modalities.

✓ Allowing for generation of **bigger INRs** and more complex data.

Andaluz.IA

DTU

Thank you!

ipeaz@dtu.dk

Code

Paper

Slides

References

- [0] Peis Aznarte, Ignacio. "Advanced Inference and Representation Learning Methods in Variational Autoencoders." (2023).
- [1] Peis, I., Koyuncu, B., Valera, I. & Frellsen, J. (2025). Hyper-Transforming Latent Diffusion Models. In International Conference on Machine Learning.
- [2] 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.
- [3] 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).
- [4] Sitzmann, V., Zollhöfer, M., & Wetzstein, G. (2019). Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems*, *32*.
- [5] Ha, David, Andrew M. Dai, and Quoc V. Le. "HyperNetworks." International Conference on Learning Representations. 2017.
- [6] Dupont, E., Whye Teh, Y.; Doucet, A.. (2022). Generative Models as Distributions of Functions. *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics, in Proceedings of Machine Learning Research* 151:2989-3015.
- [7] Dupont, E., Kim, H., Eslami, S. A., Rezende, D. J., & Rosenbaum, D. (2022, June). 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.

References

- [8] Bauer, M., Dupont, E., Brock, A., Rosenbaum, D., Schwarz, J. R., & Kim, H. (2023). Spatial functa: Scaling functa to imagenet classification and generation. arXiv preprint arXiv:2302.03130.
- [9] Koyuncu, B., Sanchez-Martin, P., Peis, I., Olmos, P. M., & Valera, I. (2023). Variational Mixture of HyperGenerators for Learning Distributions Over Functions. In Proceedings of the 40th International Conference on Machine Learning, 2023.
- [10] Chen, Yinbo, and Xiaolong Wang. "Transformers as meta-learners for implicit neural representations." European Conference on Computer Vision. Cham: Springer Nature Switzerland, 2022.
- [11] 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).
- [12]: Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., & Poole, B. Score-Based Generative Modeling through Stochastic Differential Equations. In International Conference on Learning Representations.
- [13]: Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. Advances in neural information processing systems, 33, 6840-6851.
- [14]: Song, J., Meng, C., & Ermon, S. Denoising Diffusion Implicit Models. In International Conference on Learning Representations.