DLMI2025

Diffusion models

Olivier Bernard

UNIVERSITÉ DE LYON

INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

CREATIS

olivier.bernard@insa-lyon.fr

What is the purpose of diffusion models?

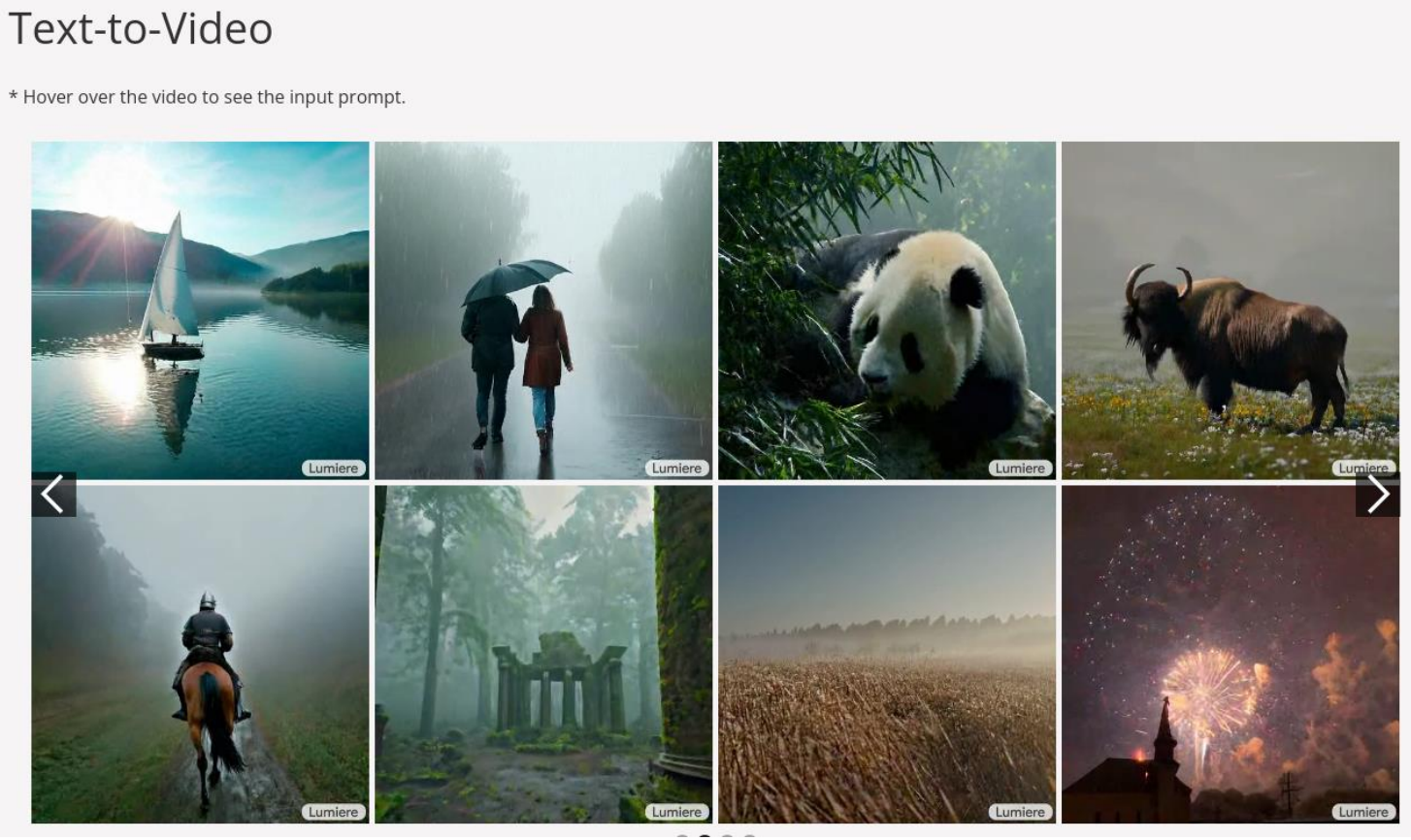
- Best current methods for synthetic image generation
- Allows generating images in a *conditioned* form
- Many software solutions, such as Midjourney, DALL-E

An Asian girl in ancient coarse linen clothes rides a giant panda and carries a wooden cage. A chubby little girl with two buns walks on the snow. High-precision clothing texture, real tactile skin, foggy white tone, low saturation, retro film texture, tranquil atmosphere, minimalism, long-range view, telephoto lens

What is the purpose of diffusion models?

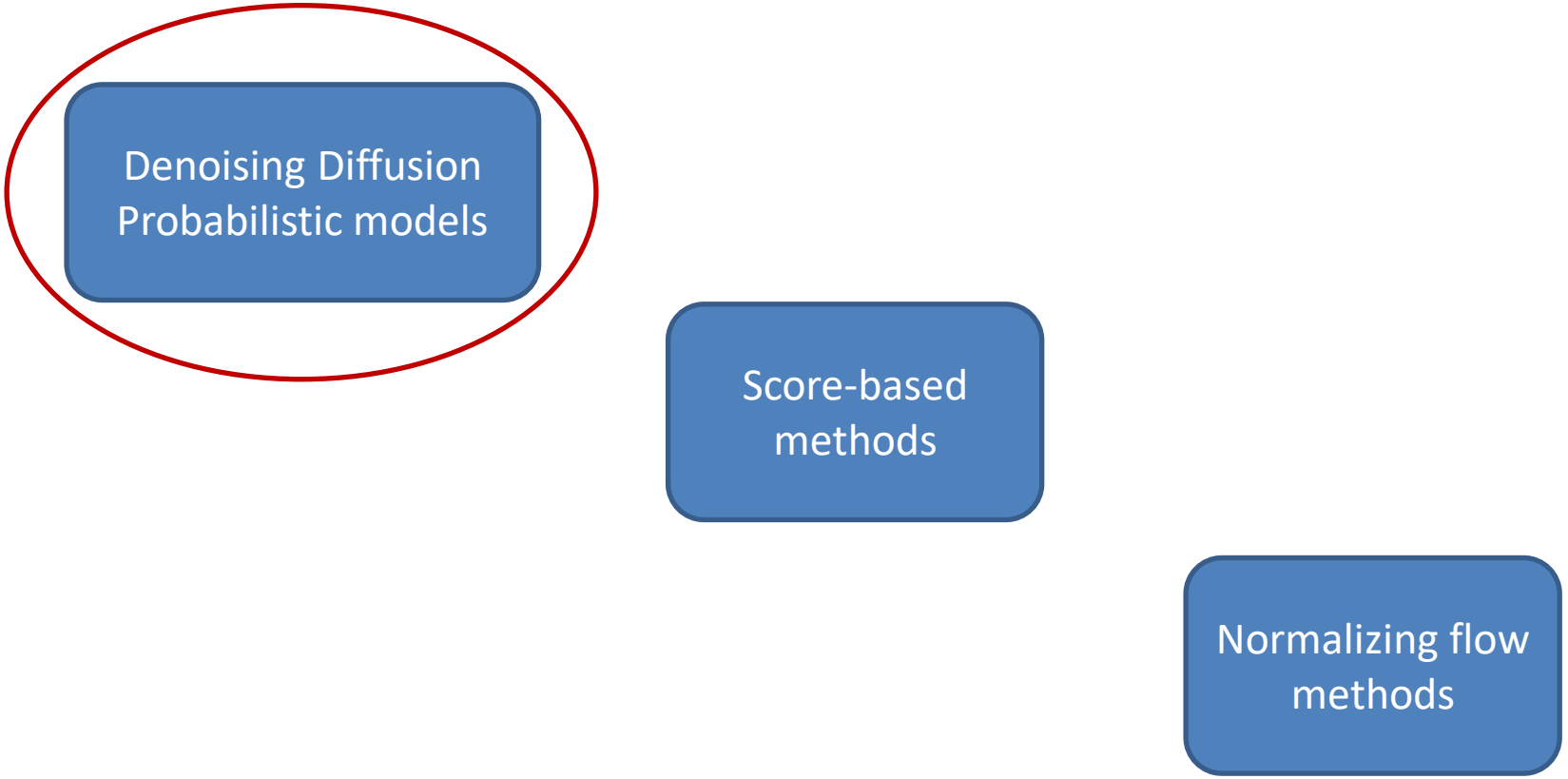
- Recent extensions for video synthesis

https://lumiere-video.github.io/#section_image_to_video



What is the purpose of diffusion models?

- Family of diffusion networks



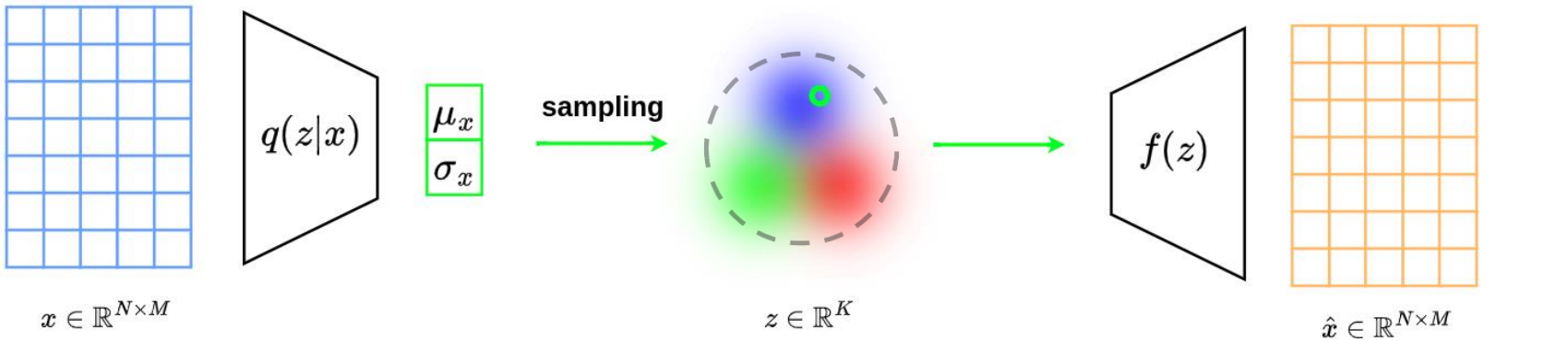
Intuition behind diffusion models

- Interpretation of the loss function

$$\text{loss} = D_{KL}\left(\mathcal{N}\left(g(x), diag\left(h(x)\right)\right), \mathcal{N}\left(0, I\right)\right) + \alpha \|x - f(z)\|^2$$

➔ $\mathcal{N}\left(g(x), diag(h(x))\right)$ imposes a local *continuity* constraint

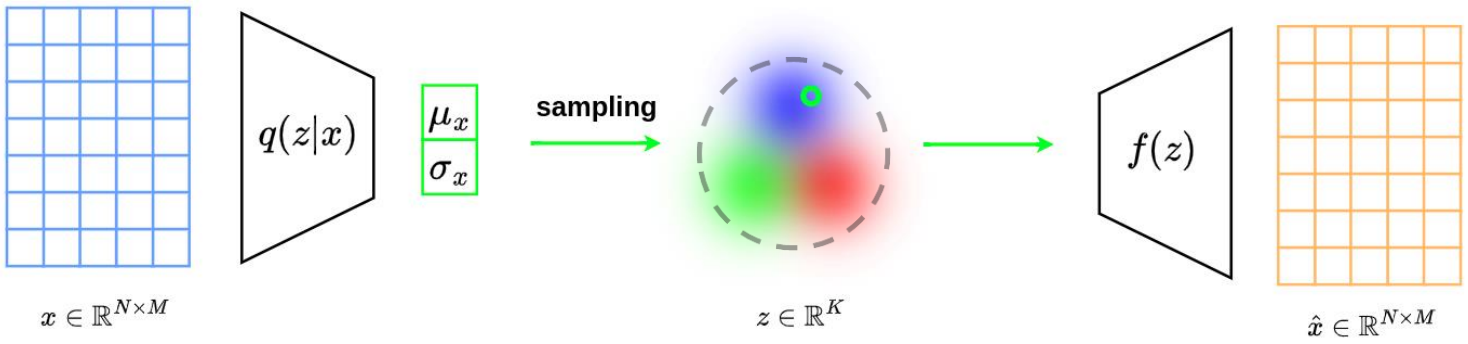
➔ $D_{KL}\left(\cdot, \mathcal{N}\left(0, I\right)\right)$ imposes a global *completeness* constraint



- Completeness is expressed as a ***soft constraint*** !

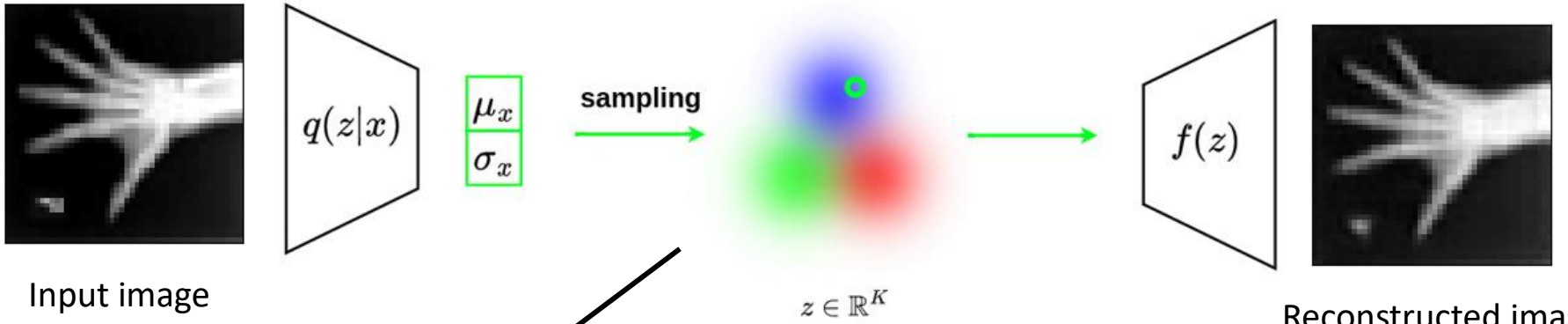
$$\text{loss} = D_{KL}\left(\mathcal{N}\left(g(x), diag\left(h(x)\right)\right), \mathcal{N}\left(0, I\right)\right) + \alpha\|x - f(z)\|^2$$

➔ $\mathcal{N}\left(g(x), diag(h(x))\right)$ and $\mathcal{N}\left(0, I\right)$ should remain close in terms of distributional distance



Sampling from the latent space $\mathcal{N}(0, I)$ does not guarantee to obtain a reconstructed image from the target distribution

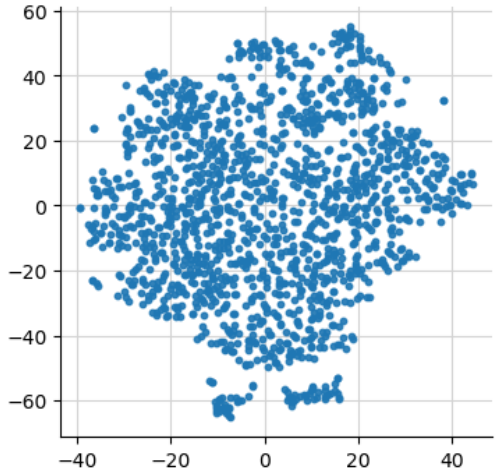
- Illustration from Mednist dataset
  - (train,valid,test) = (1491,373,223)
  - Input image size: 48x48 / latent space K=432 (compression factor around 5)



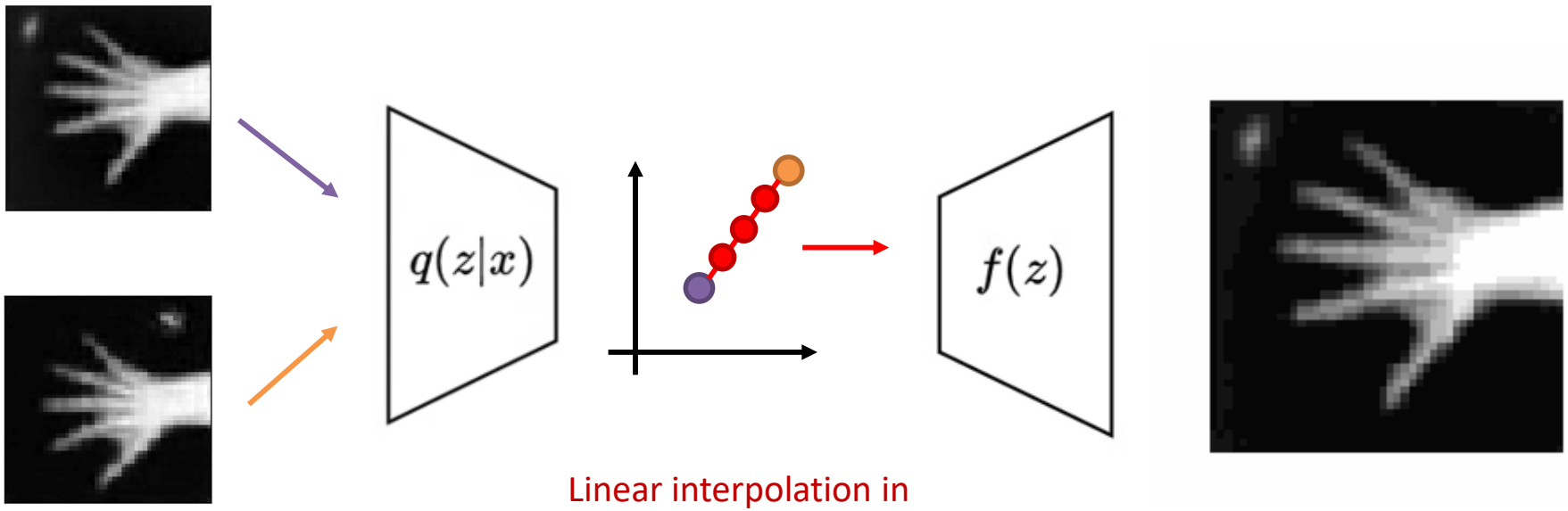
Input image

Reconstructed image

t-SNE visualization of the latent space using the training data as sample points



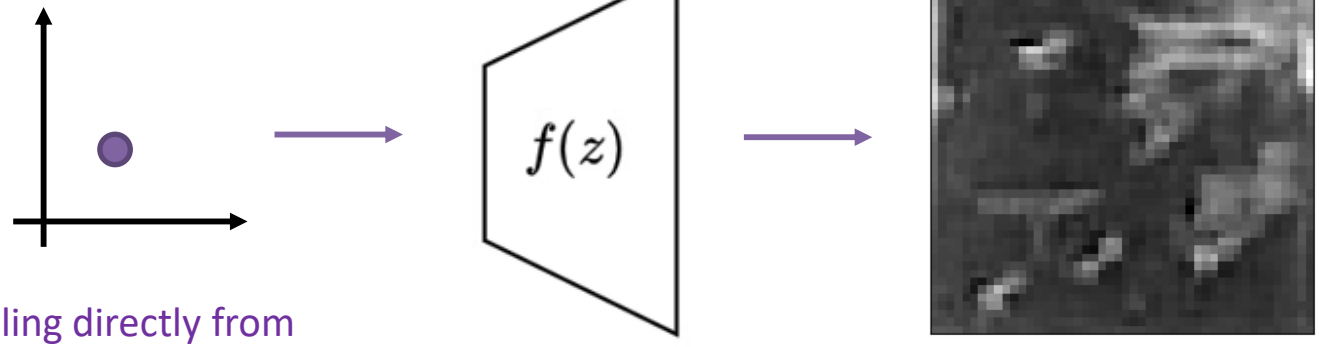
- Linear interpolation between two real images



Linear interpolation in
the latent space

- Sampling directly from the latent space

$z \in \mathbb{R}^{(K)}$ with $z_i \sim \mathcal{N}(0, I)$



Sampling directly from the latent space

A soft constraint on the latent space to remain close to $\mathcal{N}(0, I)$ is not sufficient to build generative models that effectively learn a target distribution

The denoising diffusion probabilistic models

DDPM

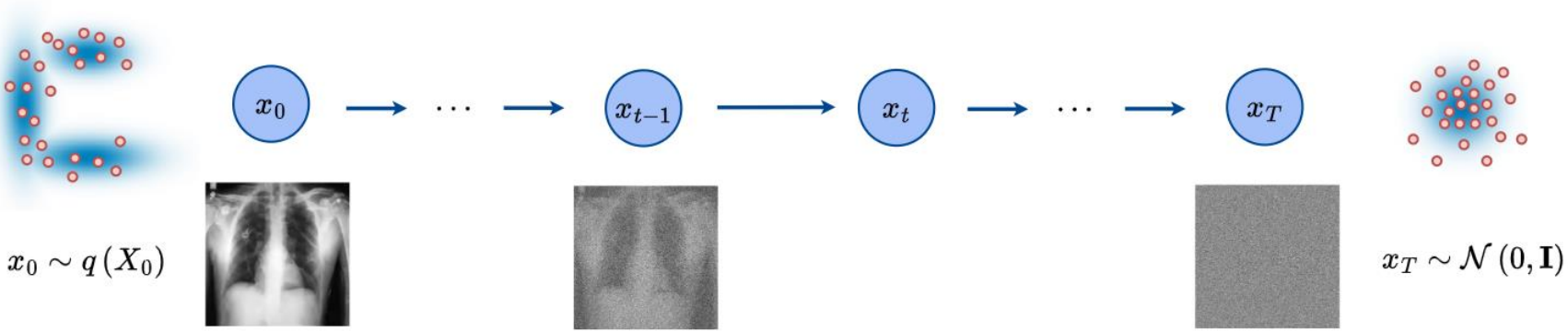
All the mathematics are described in the following blog

https://creatis-myriad.github.io/tutorials/2023-11-30-tutorial-ddpm.html

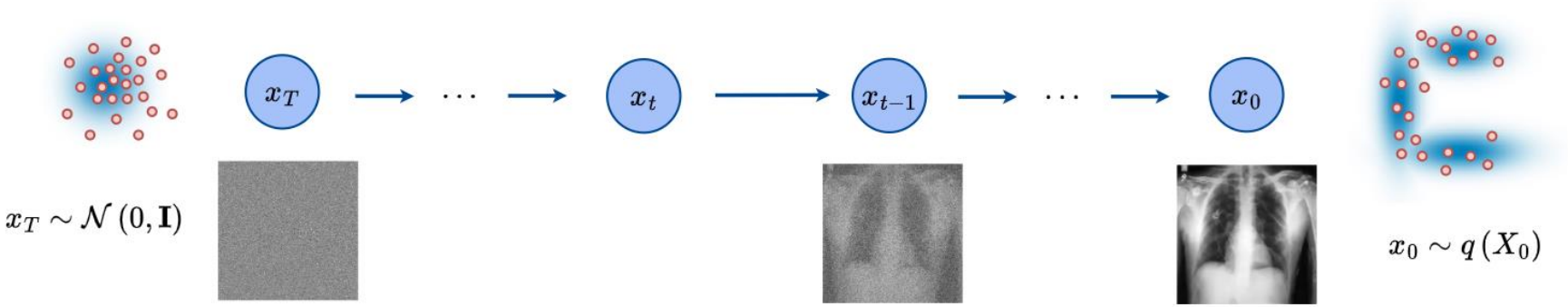
Basic idea of denoising diffusion model

How can a hard constraint be enforced to ensure a direct transformation from the latent space (modeled as a Gaussian) to the target distribution?

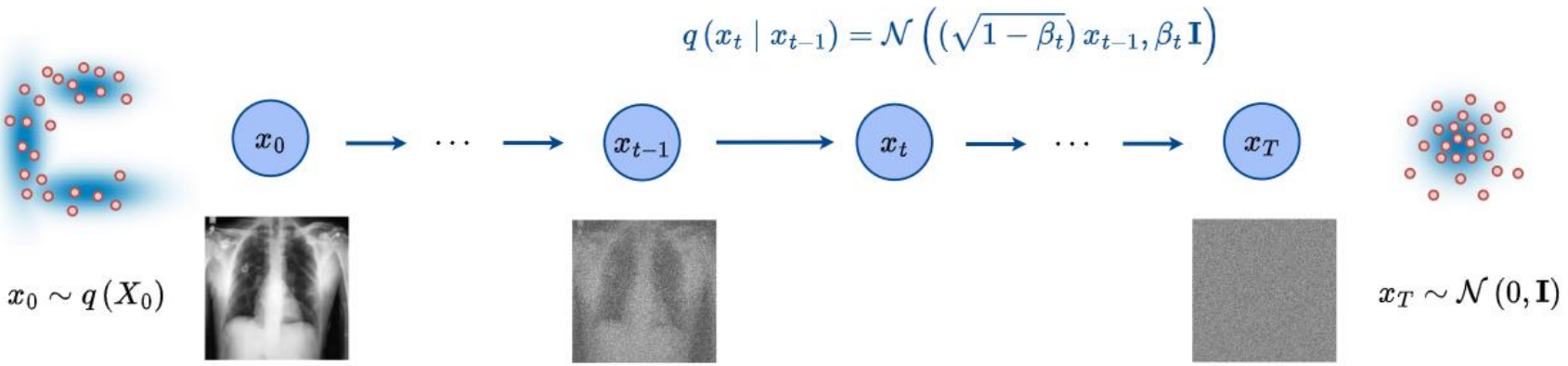
- Noising process



- Denoising process



Noising process (forward diffusion process)



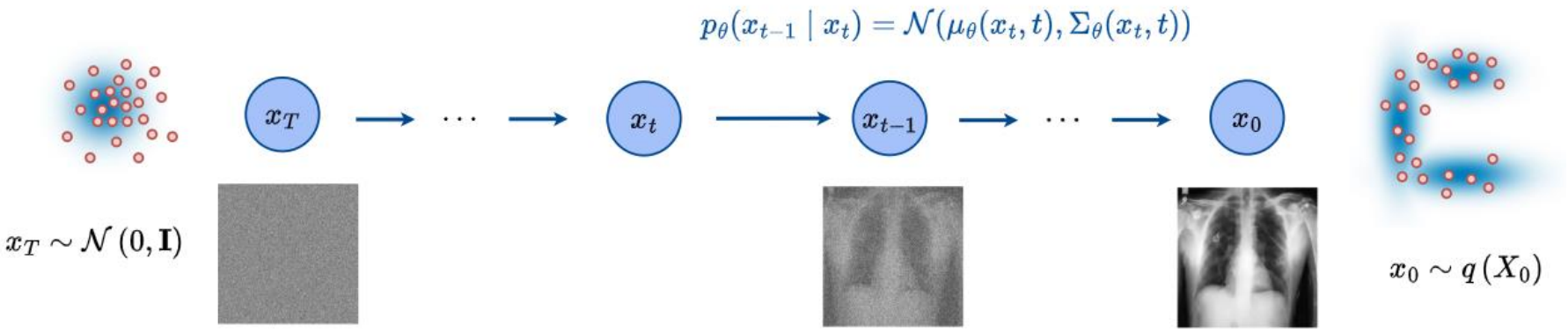
- Defined as a sequence of normal distributions

$$q(x_t \mid x_{t-1}) = \mathcal{N}\left((\sqrt{1-\beta_t})\, x_{t-1}, \beta_t \mathbf{I}\right)$$

- Forward process variances $\beta_1, \cdots, \beta_T$ with values from 0 to 1

$$\begin{aligned} &\text{if} \quad \beta_t = 0, \quad \text{then} \quad q(x_t \mid x_{t-1}) = x_{t-1} \\ &\text{if} \quad \beta_t = 1, \quad \text{then} \quad q(x_t \mid x_{t-1}) = \mathcal{N}(0, \mathbf{I}) \end{aligned}$$

Denoising process



- The denoising process p_θ is learned by the model

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}(\mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

- Knowing x_{t-1}, we need to predict μ_θ and Σ_θ

  - $\Sigma_\theta = \sigma_t^2 I$ with $\sigma_t = \beta_t$ for simplification purposes
  - Predicting μ_θ involves estimating the added noise ε_t from x_{t-1} to x_t

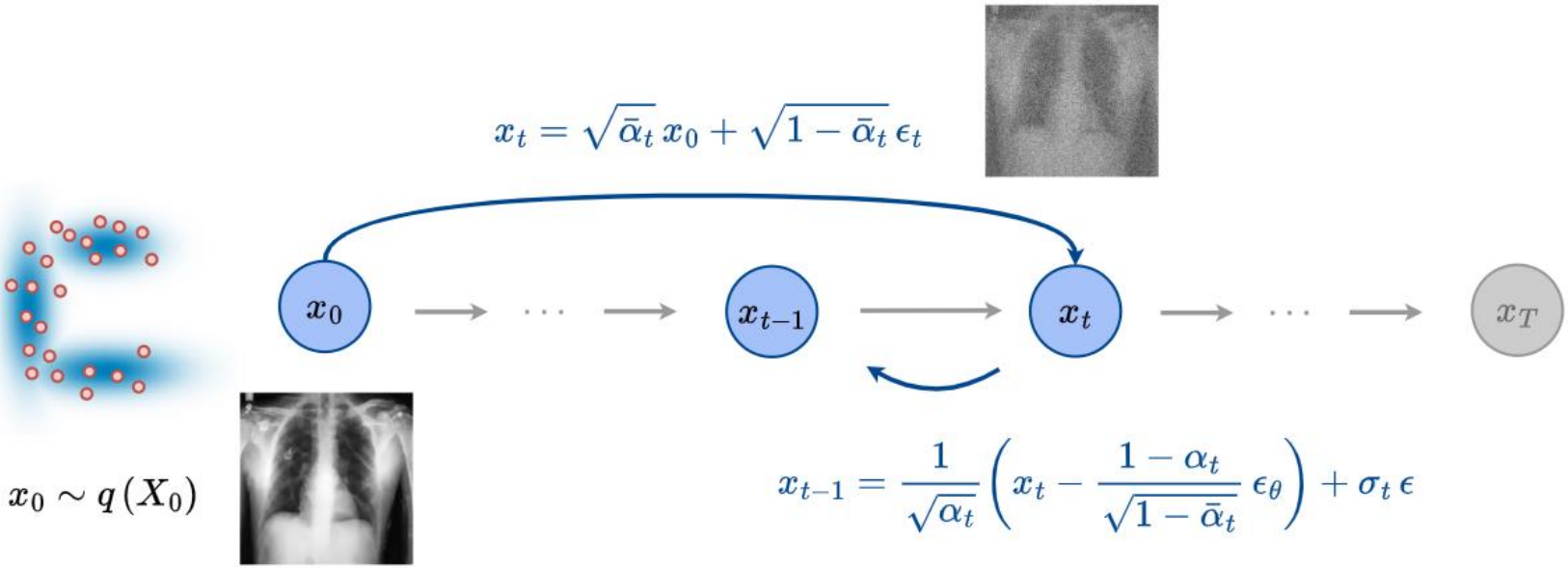
Training procedure

- Choose a random step $t \in \{0, \cdots, T\}$
- Add t steps of noise to our input image x_0, and obtain a noisy image x_t

$$x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1 - \bar{\alpha}_t}\, \epsilon_t$$

$$\begin{cases} \alpha_t = 1 - \beta_t \\ \bar{\alpha}_t = \prod_{k=1}^{t} \alpha_k \end{cases} \qquad \begin{cases} \epsilon_t = \mathcal{N}(0, \mathbf{I}) \\ \text{added noise from } x_{t-1} \text{ to } x_t \end{cases}$$

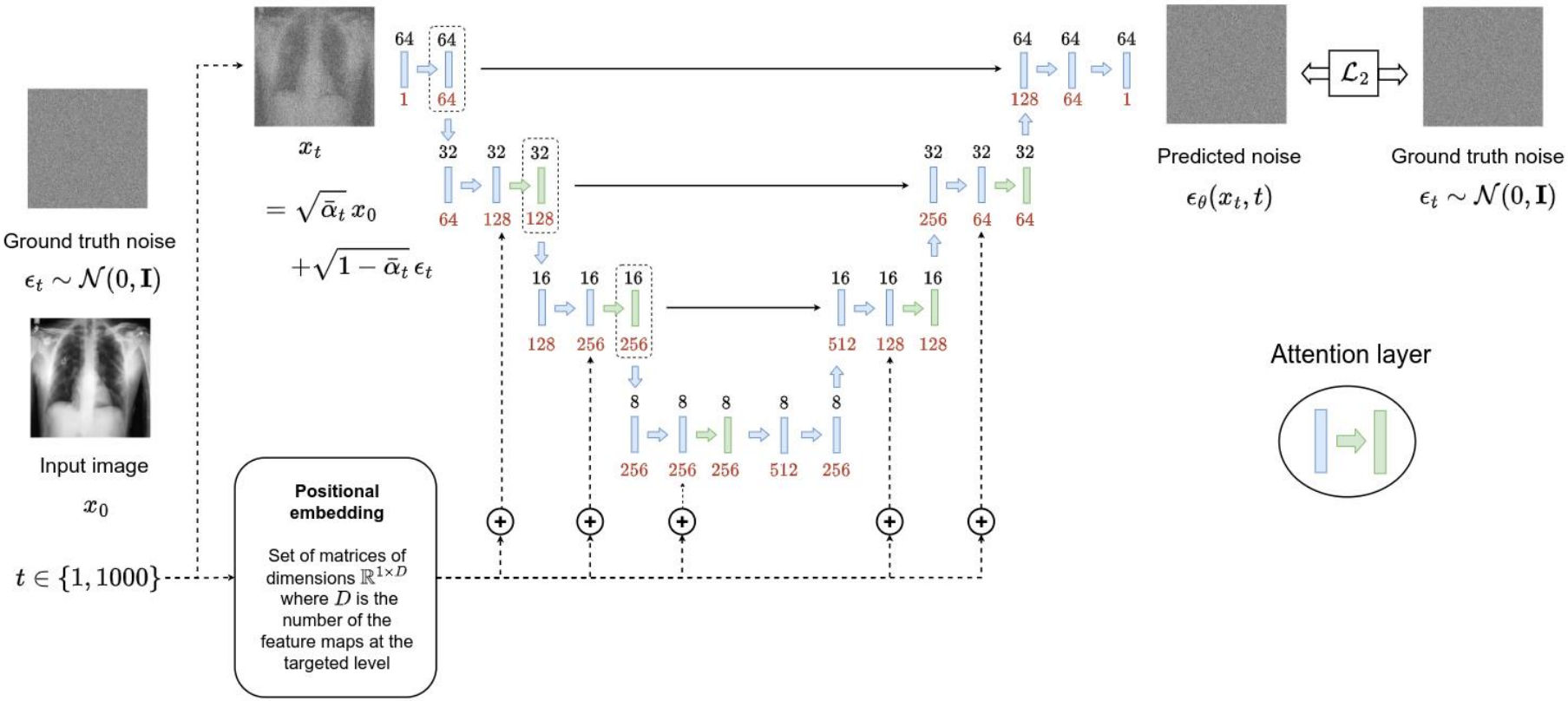
- A U-Net model is trained to predict the noise pattern ε_θ that needs to be subtracted to x_t to predict a slightly denoised x_{t-1}



Architecture

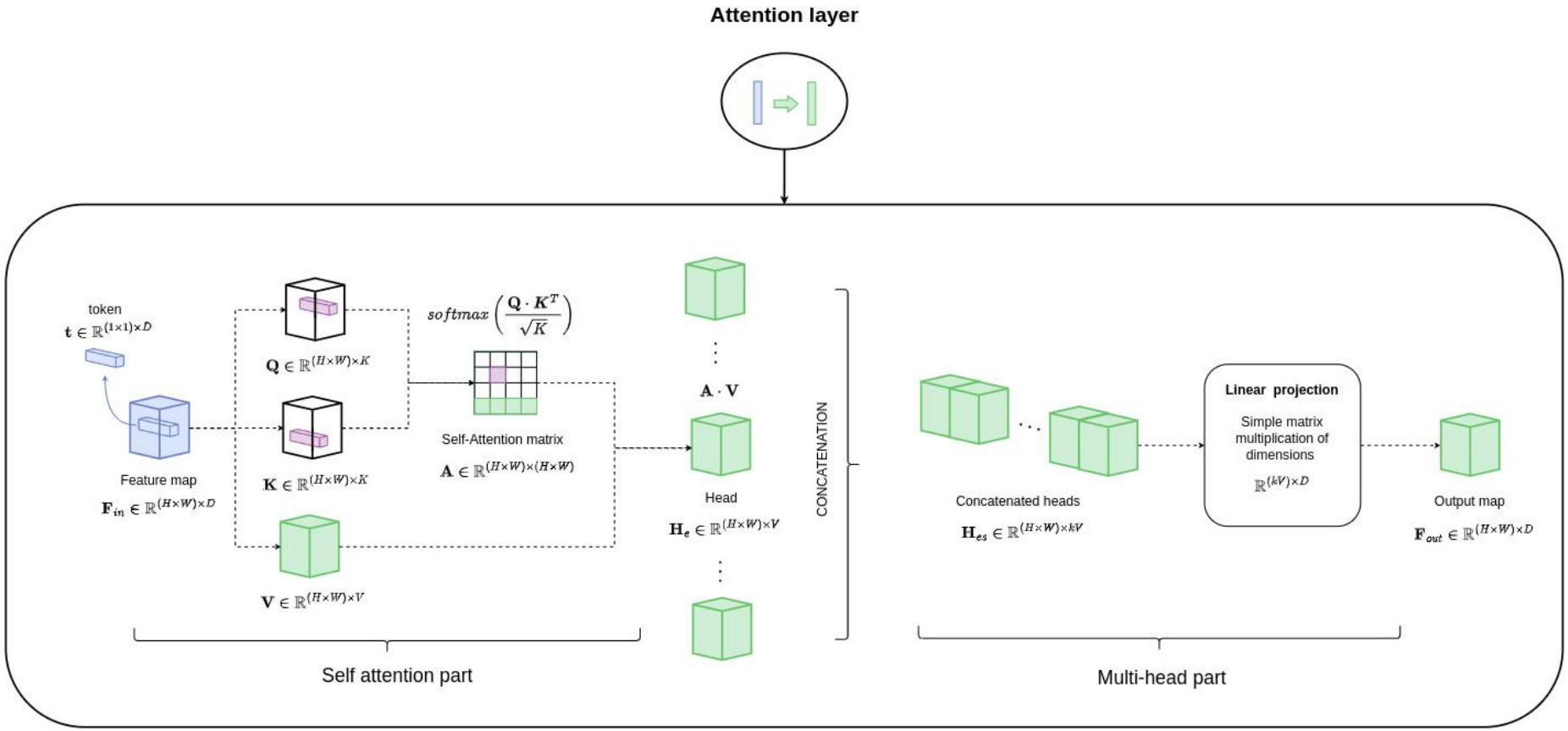
Standard U-Net with attention layers and position encoding to integrate temporal information

➔ Integration of t is necessary because the added noise varies over time

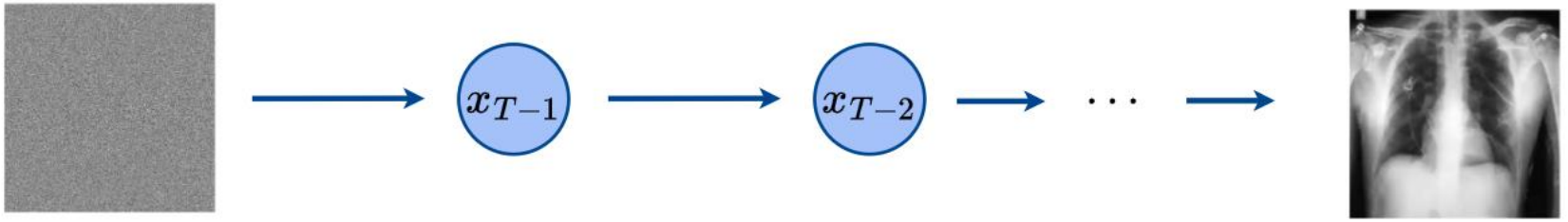


Architecture

➔ Attention layer



Inference: generation of synthetic data



$x_T \sim \mathcal{N}(0, \mathbf{I})$

x_0

- Generate a random image $x_T \sim \mathcal{N}(0, I) \in \mathbb{R}^{N \times M}$
- At each step from T to 0, use the U-Net model to compute

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(x_t, t) \right) + \sigma_t \epsilon$$

U-Net

with $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ and

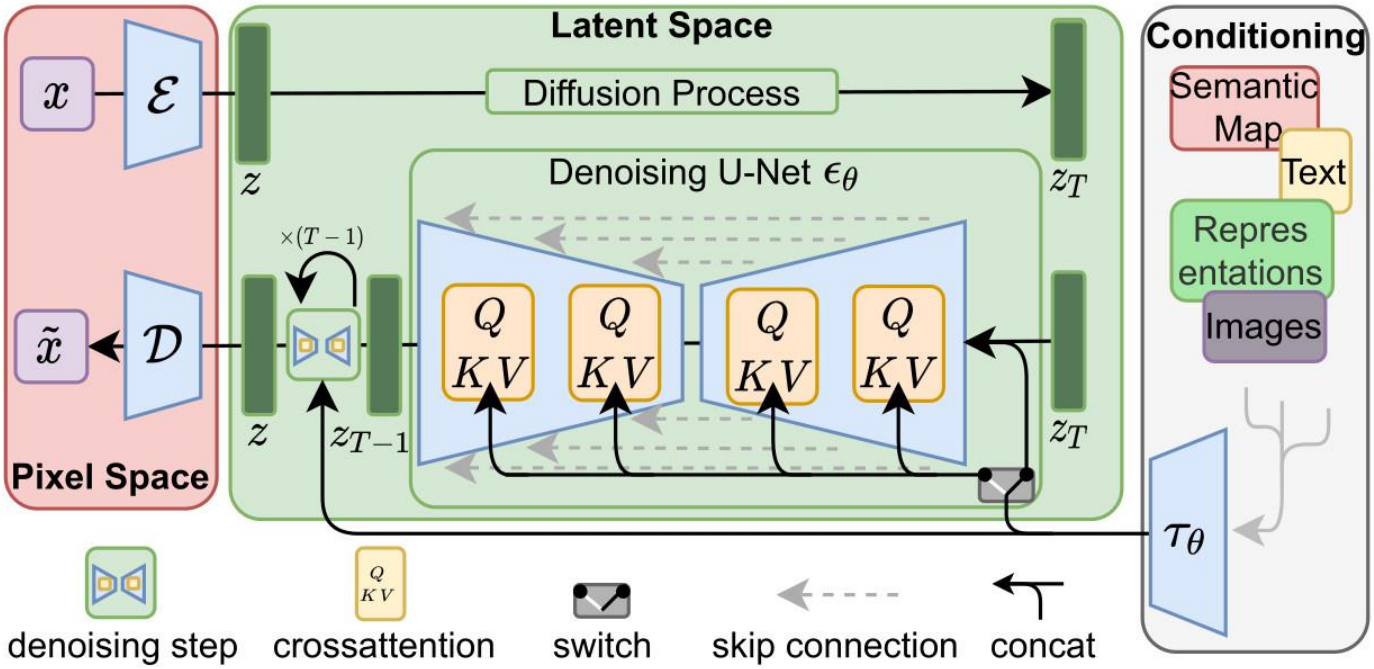
$$\begin{cases} \alpha_t = 1 - \beta_t \\ \bar{\alpha}_t = \prod_{k=1}^{t} \alpha_k \end{cases}$$

Practical application

Latent diffusion models

Latent diffusion model (LDM)

- VAE is learned independently of DDPM and its architecture is fixed
  - Efficiently reduce the dimensionality of the input space
  - Efficiently initiate the Gaussian diffusion process
- LDM architecture



- Properties

Parameters	LDM – 256×256
z dimensions	$\mathbf{64} \times \mathbf{64} \times 3$
Diffusion steps	$\mathbf{1000}$
Noise scheduler (β_t)	linear
Number of parameters	$\mathbf{274}$ Million
Channels	$\mathbf{224}$
Channel multiplier	$\mathbf{1, 2, 3, 4}$
Levels for attention	$\mathbf{2, 3, 4}$
Number of head	$\mathbf{1}$
Batch size	$\mathbf{48}$
Iterations	$\mathbf{410}$ k
Learning rate	$\mathbf{9.6}\ e^{-5}$

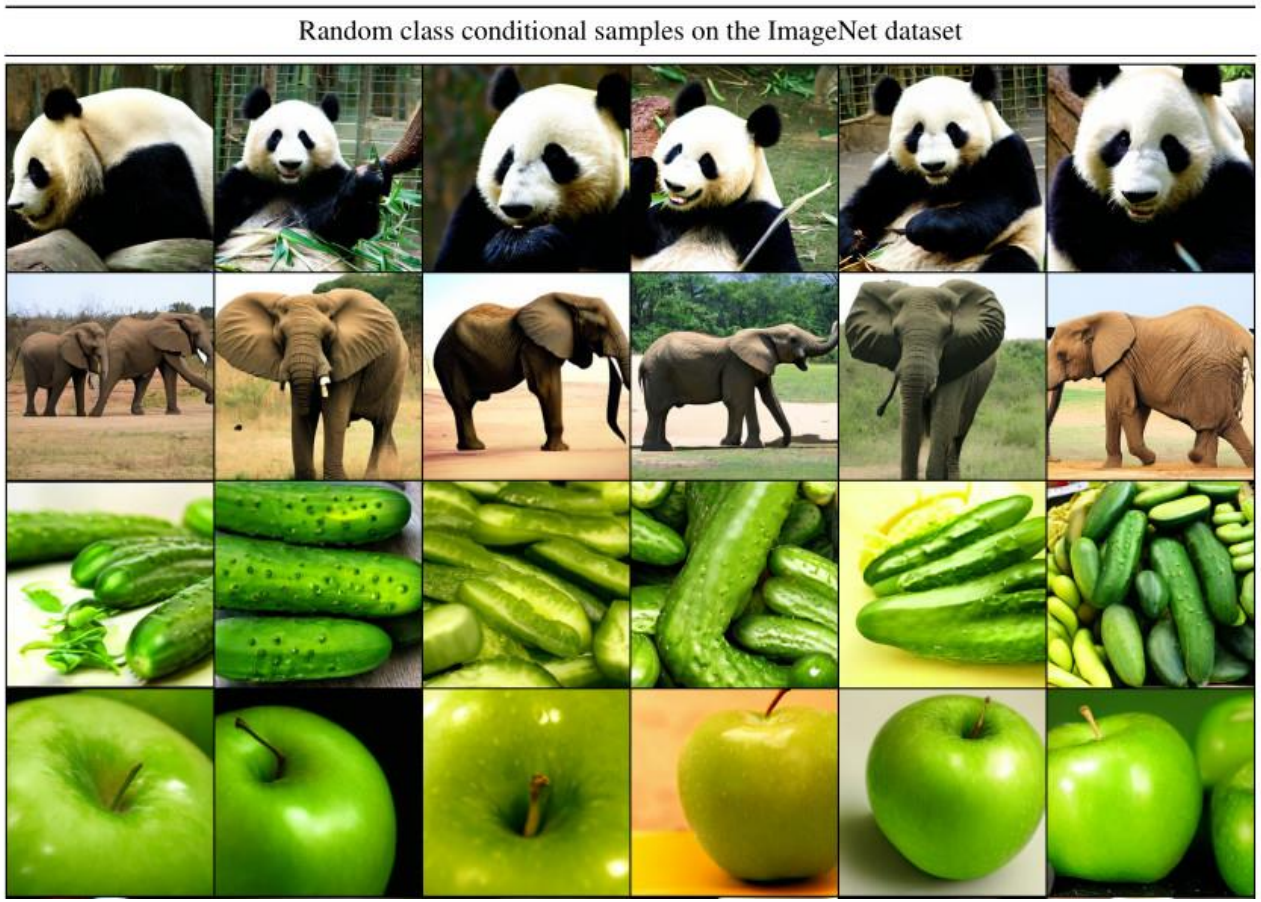
Latent diffusion model (LDM)

- Random generation of synthetic images *without conditioning* learned from the CelebA-HQ database

Random samples on the CelebA-HQ dataset

Latent diffusion model (LDM)

- Random generation of synthetic images *with conditioning on the class* learned from the ImageNet database



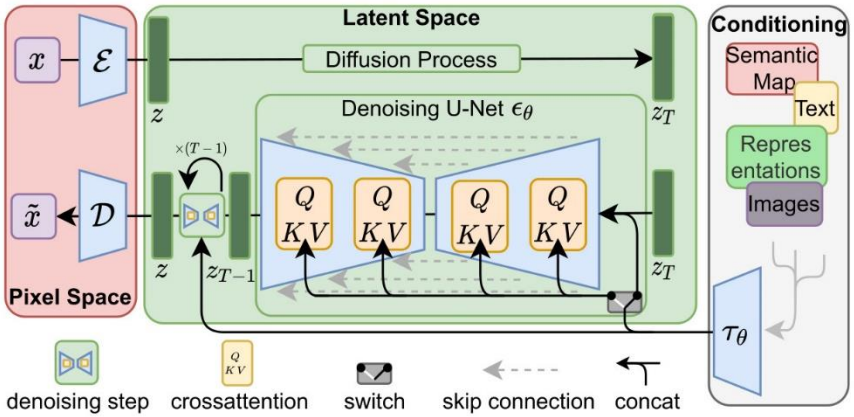
Latent diffusion model (LDM)

- Random generation of synthetic images *with conditioning on masks* learned from the Flickr-landscapes database

Medical applications

Denoising

Data augmentation

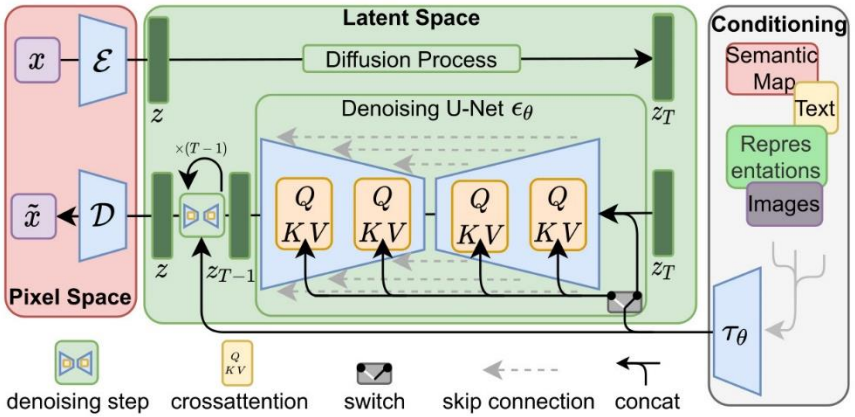


Segmentation

Anomaly detection

Denoising

Data augmentation



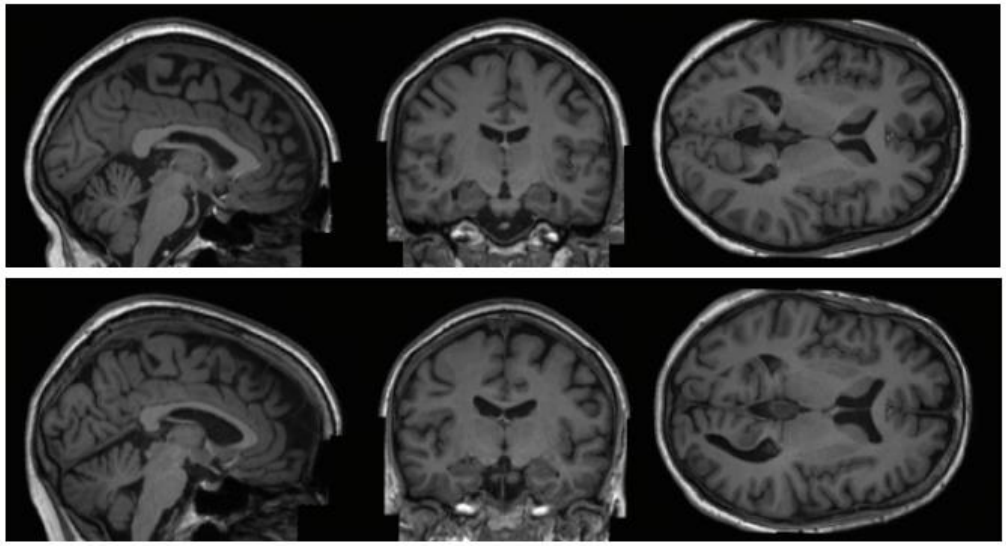
Segmentation

Anomaly detection

Diffusion models for data augmentation

- Synthetic dataset generation for brain MR volumes [Walter et al., MICCAI workshop 2022]

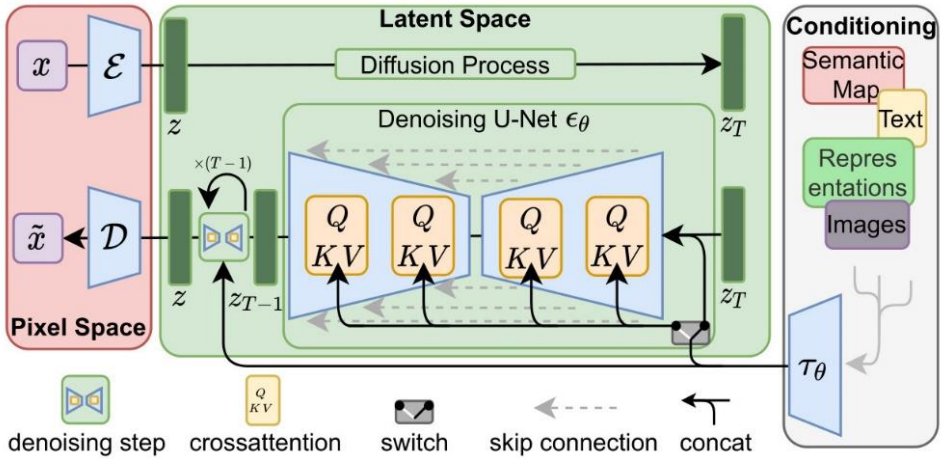
- UK Biobank dataset
  - 3D MR volumes (T1w)
  - Training: 31,740 patients
  - with covariables: age (44 to 82 years), gender (53% women), brain structure volumes
  - Quality of synthetic data measured using FID: Fréchet Inception Distribution



Diffusion models for data augmentation

- VAE
  - 3D convolutions
  - Latent space dimension: 20 x 28 x 20

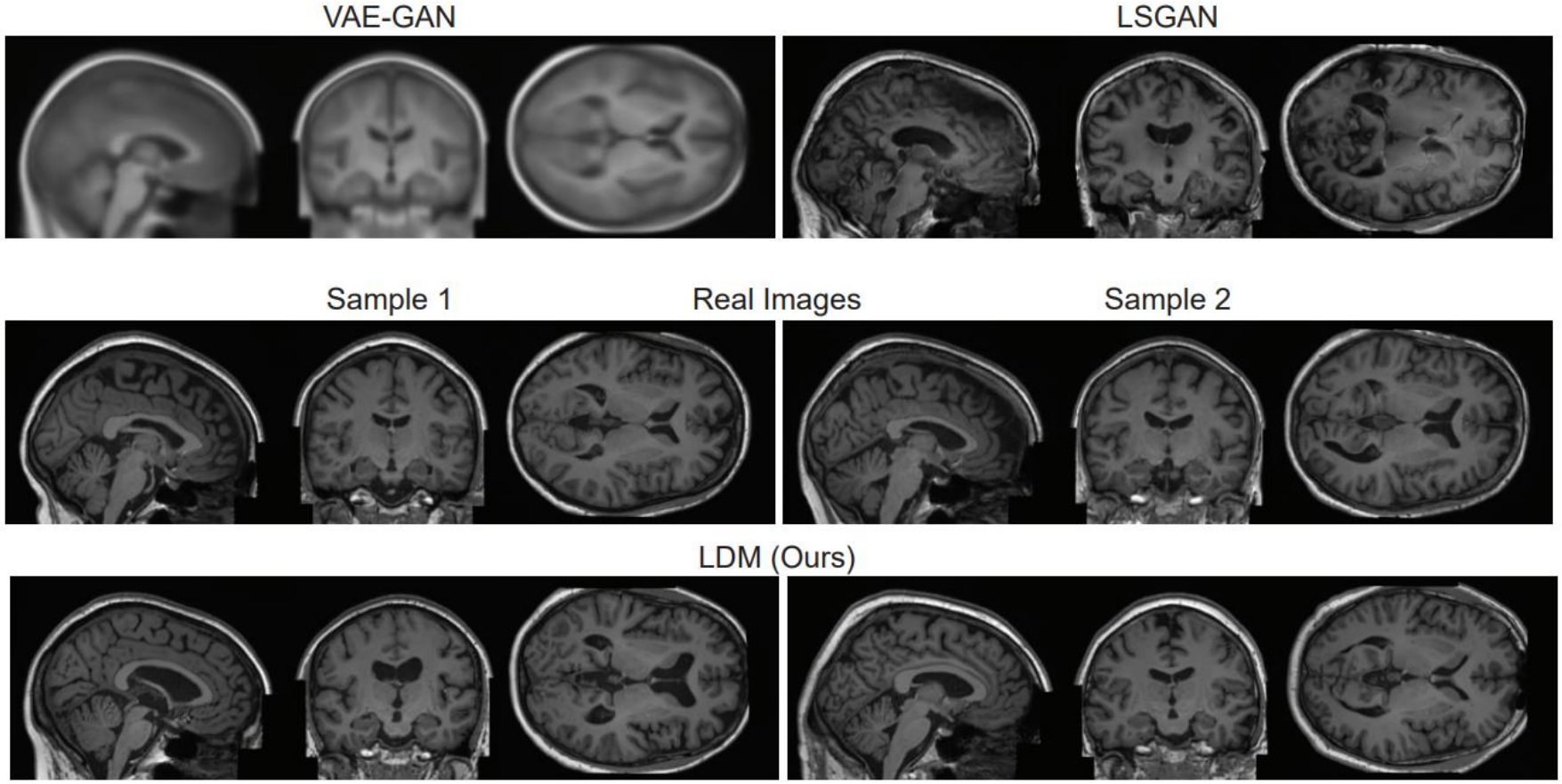
- DDPM
  - 3D convolutions
  - T=1000 time steps
  - Conditioning: vector encoding of each covariable



Diffusion models for data augmentation

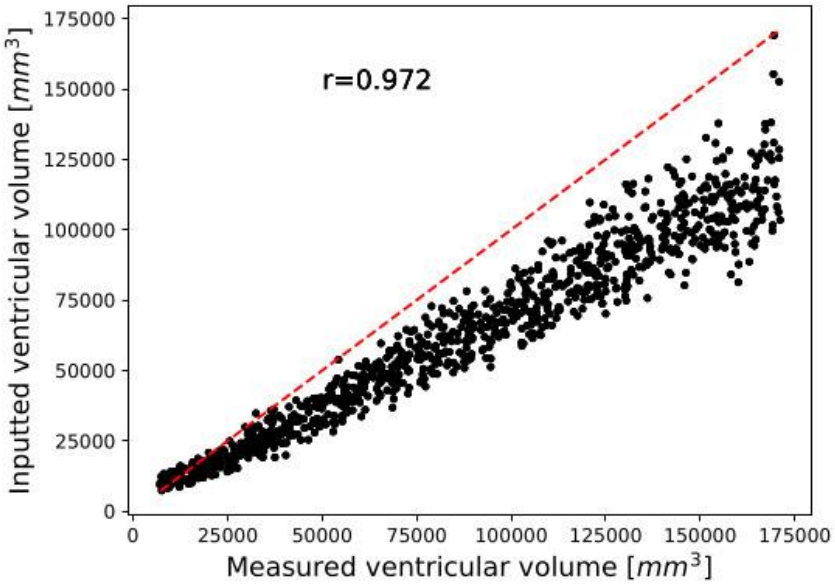
- Results
  - FID: generated from 1,000 samples drawn from each of the two distributions to be compared

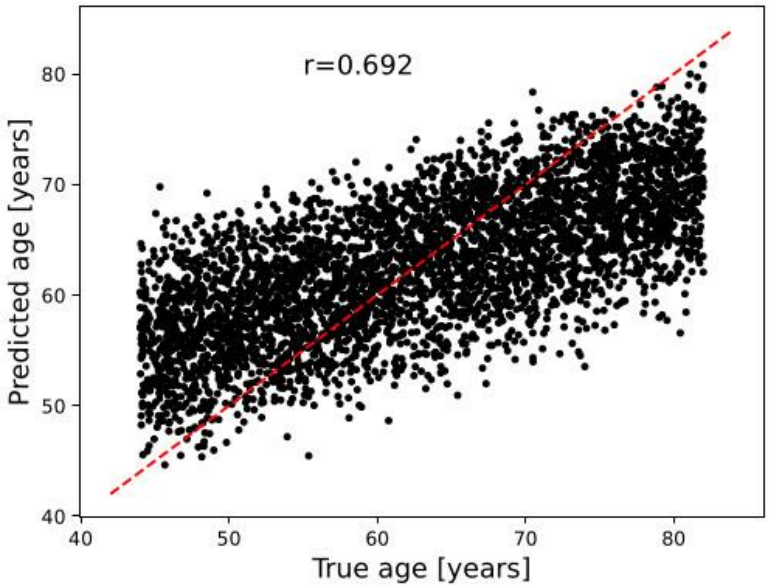
	FID ↓
LSGAN	0.0231
VAE-GAN	0.1576
LDM	**0.0076**
Real images	0.0005



Diffusion models for data augmentation

- Results
  - SynthSeg model was used to automatically measure brain volumes from synthetic data
  - A 3D CNN trained from the UK biobank was used to automatically predict the age from the synthetic data

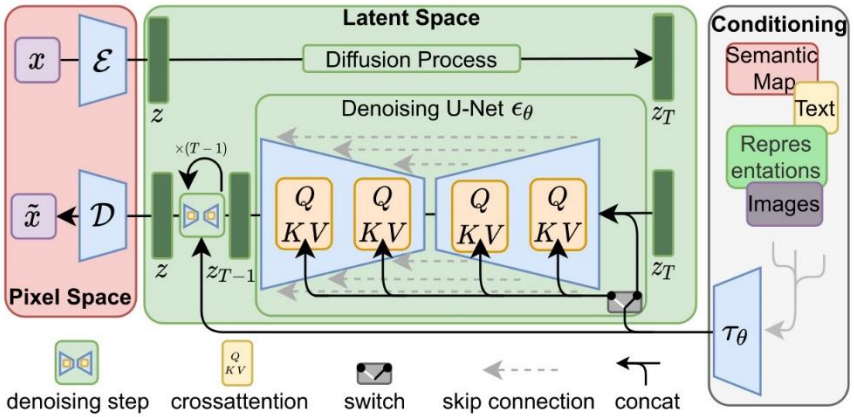




- Synthetic dataset of 100,000 human brain was generated and made publicly available with the conditioning information
- Promote data sharing with privacy guarantees

Denoising

Data augmentation



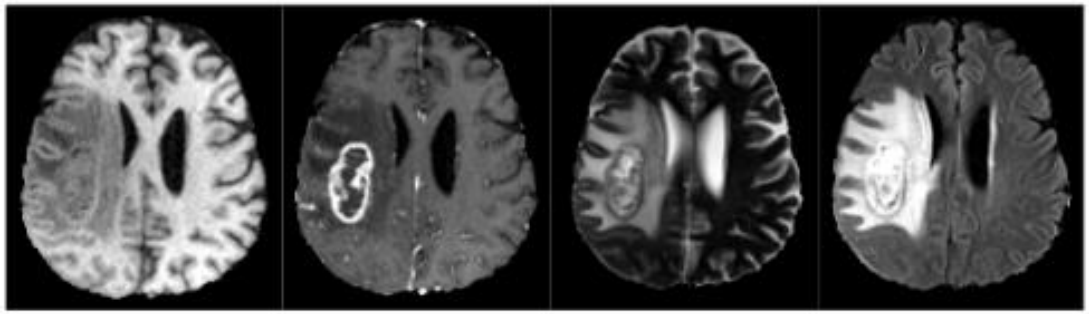
Segmentation

Anomaly detection

Diffusion models for image segmentation

- Segmentation of tumors from MR images [Wolleb et al., MIDL 2022]

- BRATS2020 dataset
  - 4 different MR sequences per patient (T1, T2, T1ce, FLAIR)
  - Training: 332 patients with 3D volumes sequences => 16,998 2D images
  - Testing: 37 patients with 3D volumes sequences => 1,082 2D images

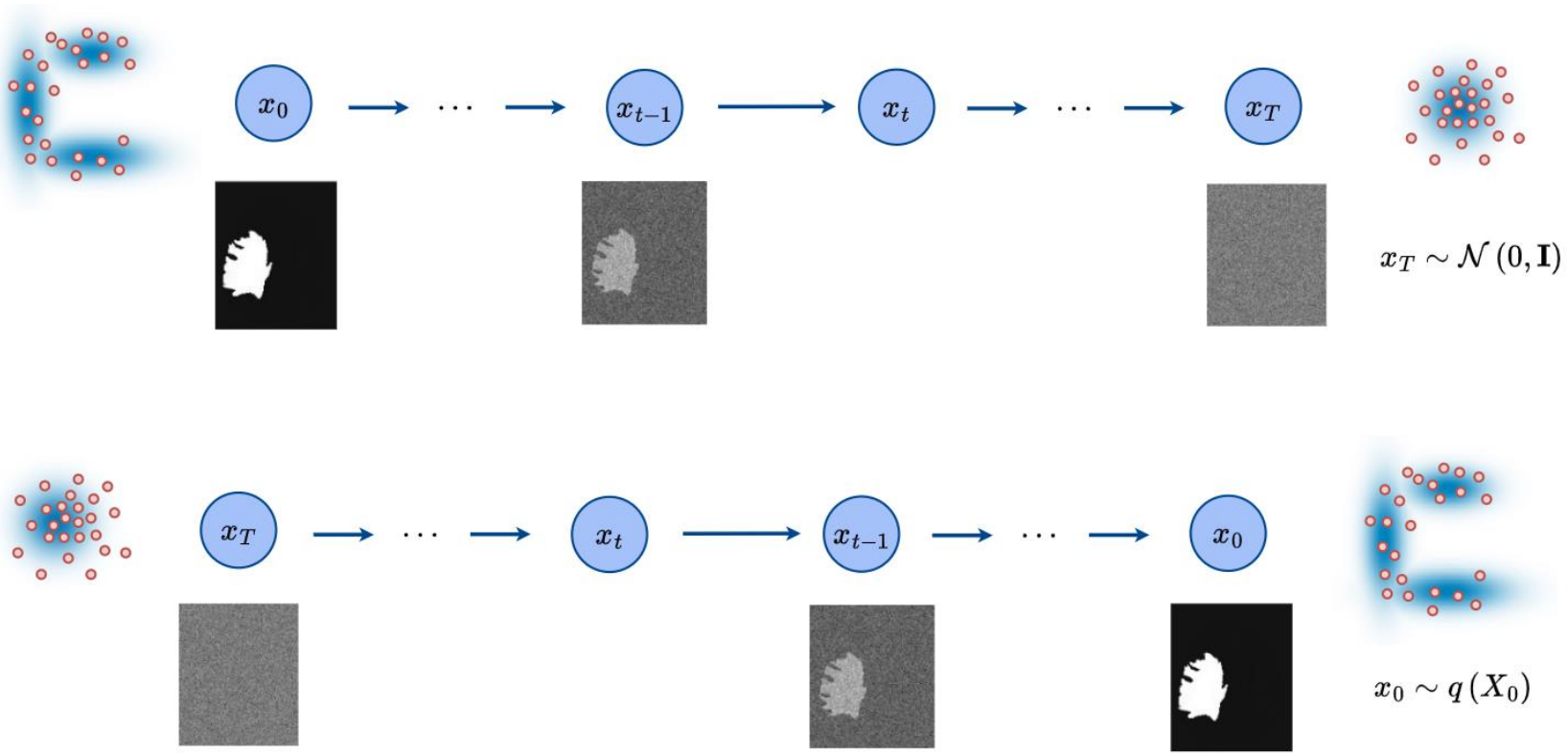


4 MR inputs per patient (T1, T2, T1ec, FLAIR)

Mask output

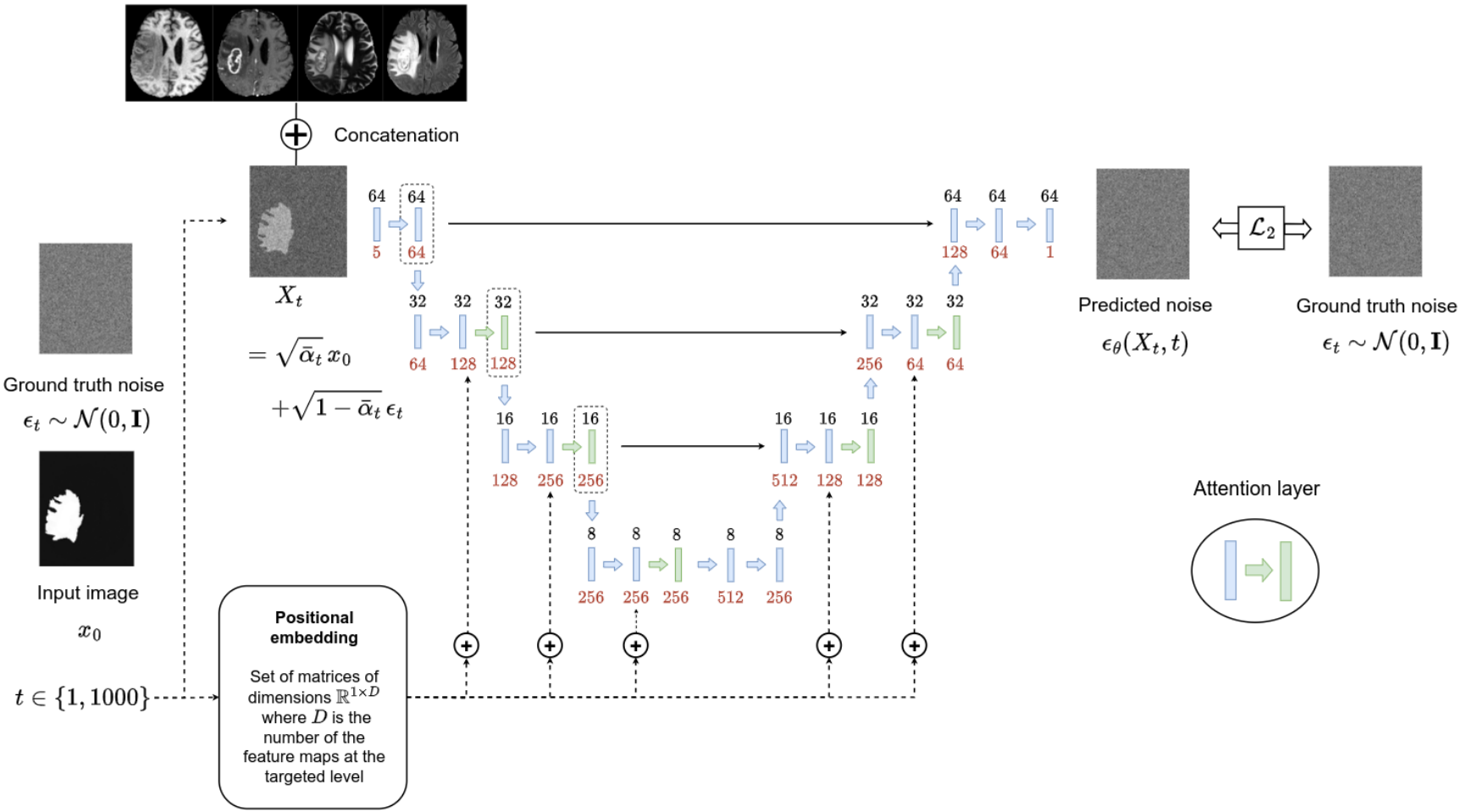
Diffusion models for image segmentation

- Learn the underlying distribution of tumor segmentation masks



Diffusion models for image segmentation

- Conditioning with the 4 MR images using concatenation scheme



Diffusion models for image segmentation

- At inference time: modelling of the segmentation uncertainty

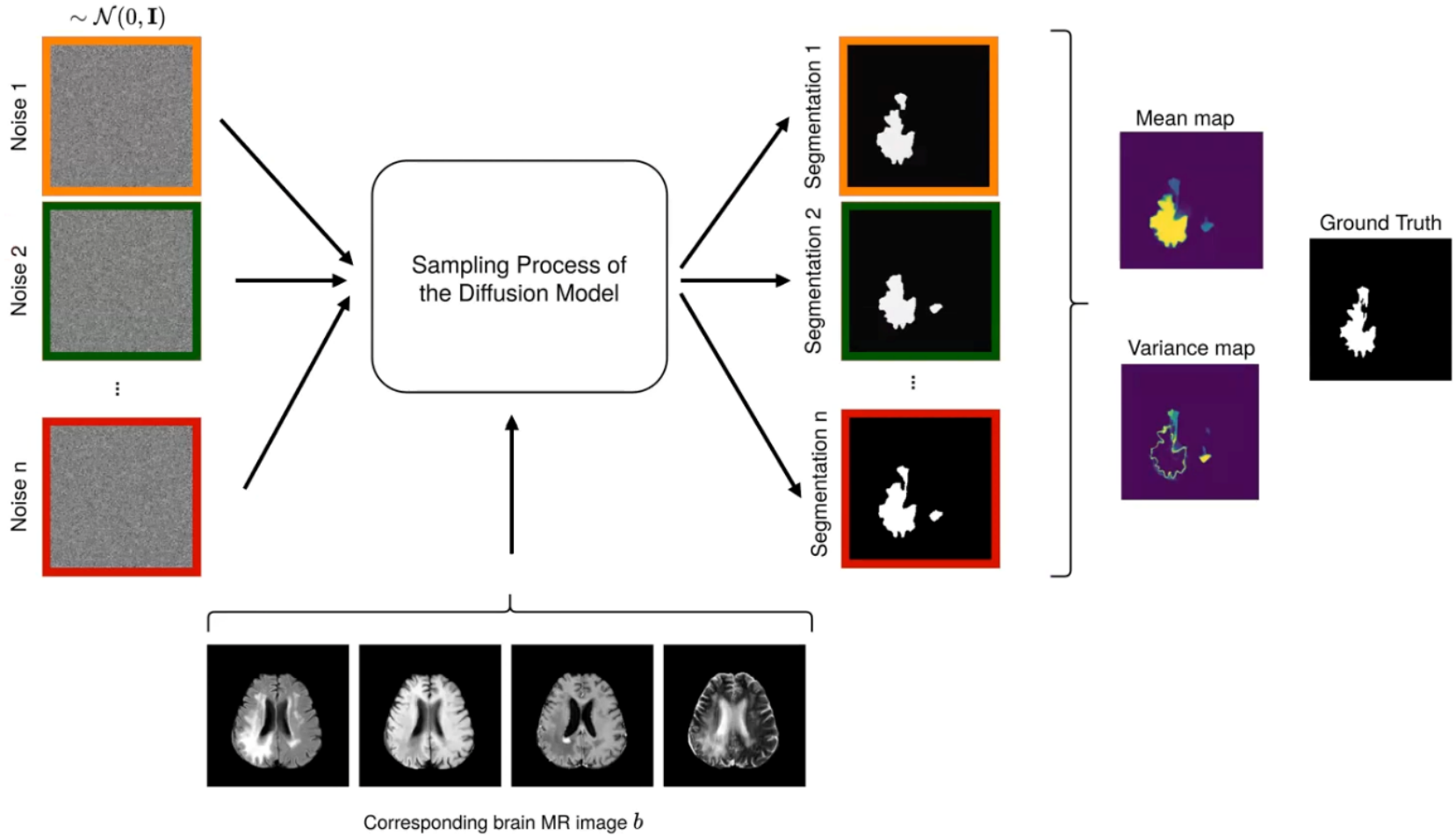


Illustration taken from https://www.youtube.com/watch?v=US9CzPrT2H8

Diffusion models for image segmentation

- Results

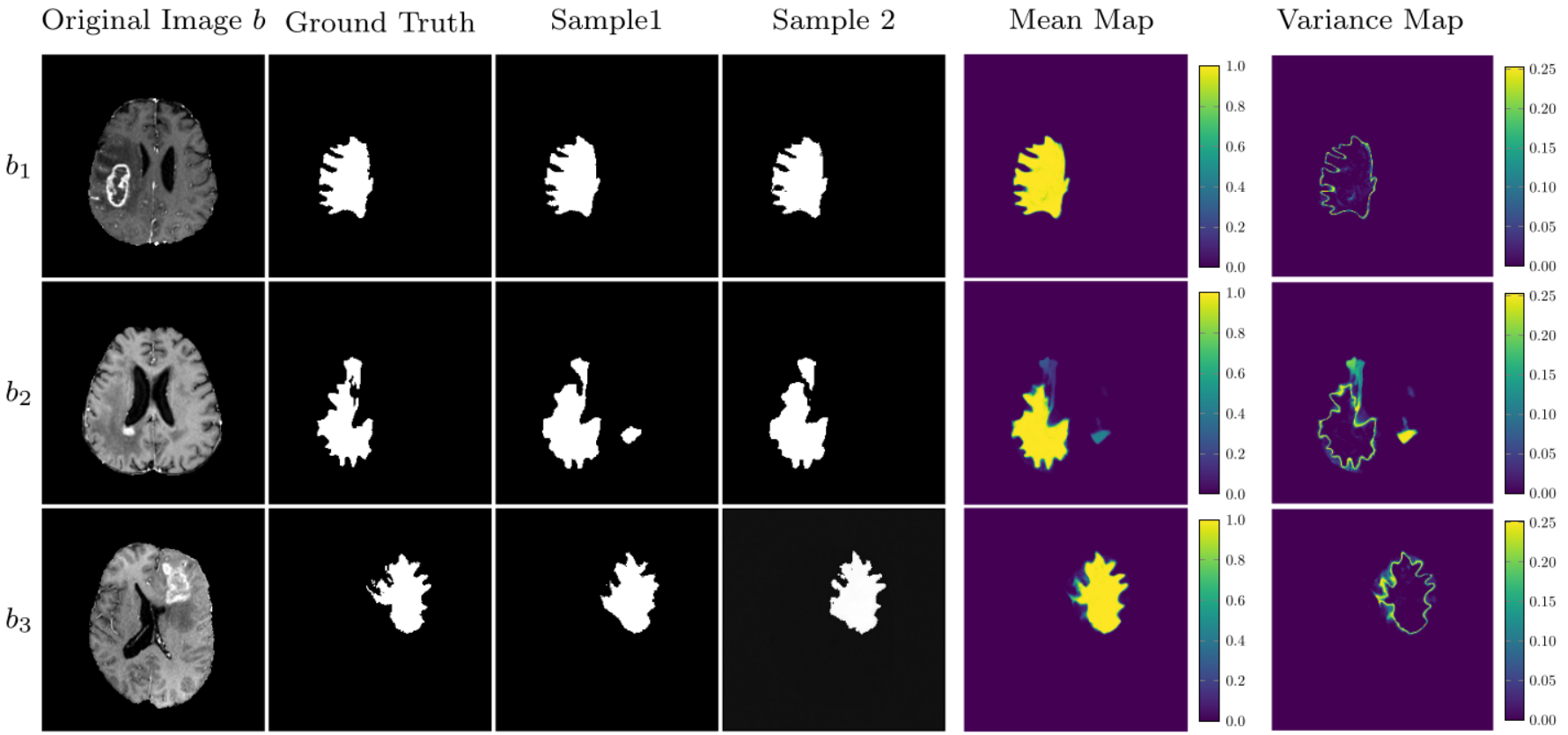
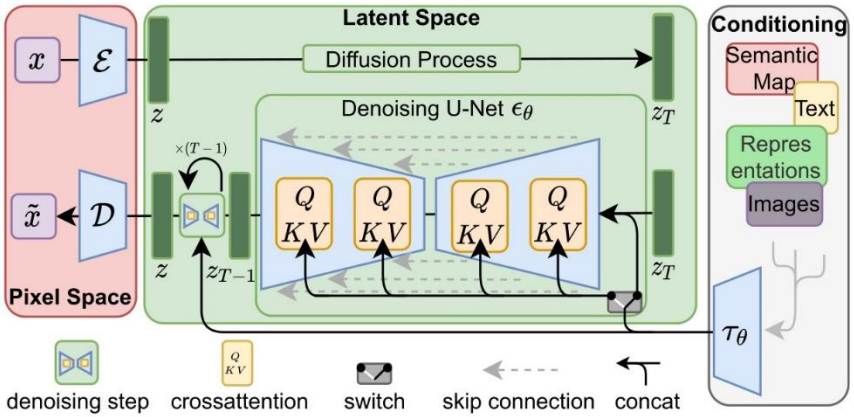


Illustration taken from https://arxiv.org/pdf/2112.03145

Denoising

Data augmentation



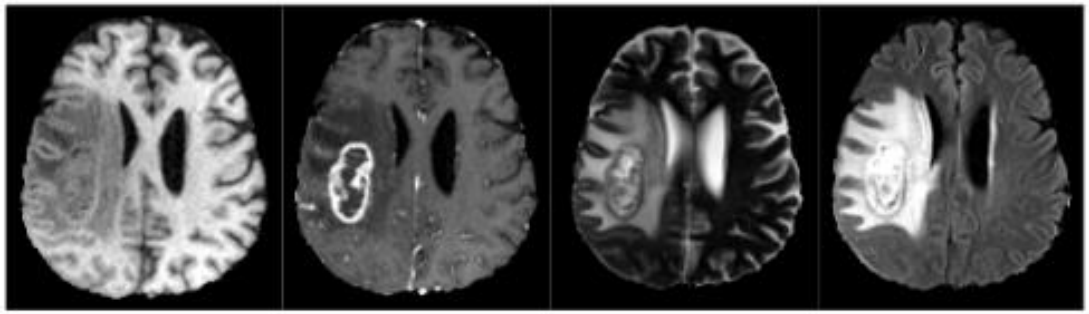
Segmentation

Anomaly detection

Diffusion models for anomaly detection

- Anomaly detection from MR images [Wolleb et al., MICCAI 2024]

- BRATS2020 dataset
  - 4 different MR sequences per patient (T1, T2, T1ce, FLAIR)
  - Training: 332 patients with 3D volumes sequences => 16,998 2D images
  - 5,598 healthy 2D slices (without tumor) / 10,607 disease 2D slices

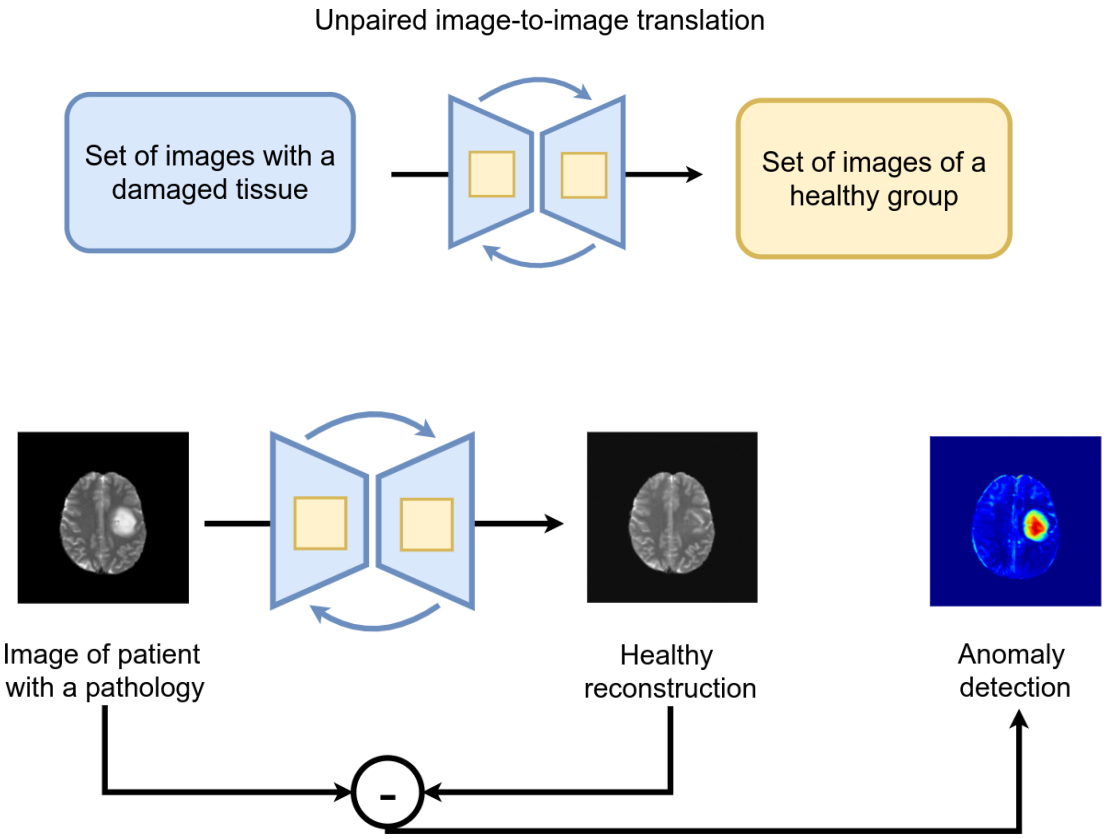


4 MR inputs per patient (T1, T2, T1ec, FLAIR)

Mask output

Diffusion models for anomaly detection

- General idea



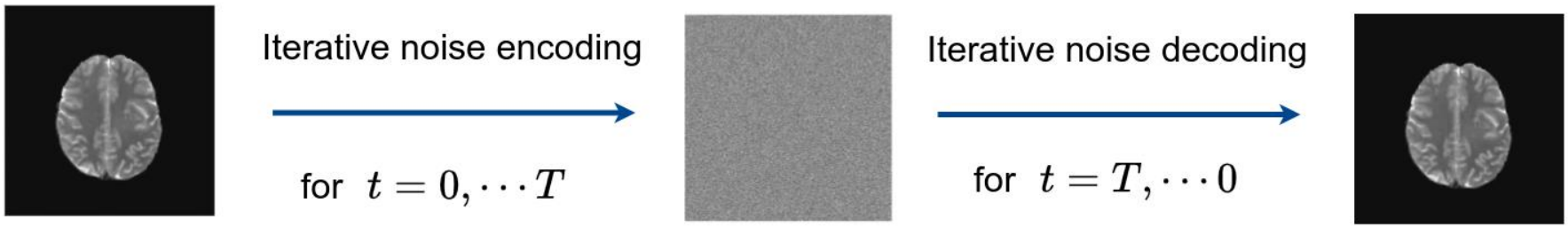
How to preserve spatial anatomical information using a diffusion process?

- Denoising Diffusion Implicit Models (DDIM)
  - Reformulation of the diffusion process
  - Remove the random component $\sigma_t \epsilon$

$$x_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \left(\frac{x_t - \sqrt{1-\bar{\alpha}_t} \cdot \epsilon_\theta(x_t, t)}{\sqrt{\bar{\alpha}_t}} \right) + \sqrt{1-\bar{\alpha}_{t-1}} \cdot \epsilon_\theta(x_t, t)$$

$$x_{t+1} = x_t + \sqrt{\bar{\alpha}_{t+1}} \left[\left(\sqrt{\frac{1}{\bar{\alpha}_t}} - \sqrt{\frac{1}{\bar{\alpha}_{t+1}}} \right) x_t + \left(\sqrt{\frac{1}{\bar{\alpha}_{t+1}} - 1} - \sqrt{\frac{1}{\bar{\alpha}_t} - 1} \right) \epsilon_\theta(x_t, t) \right]$$

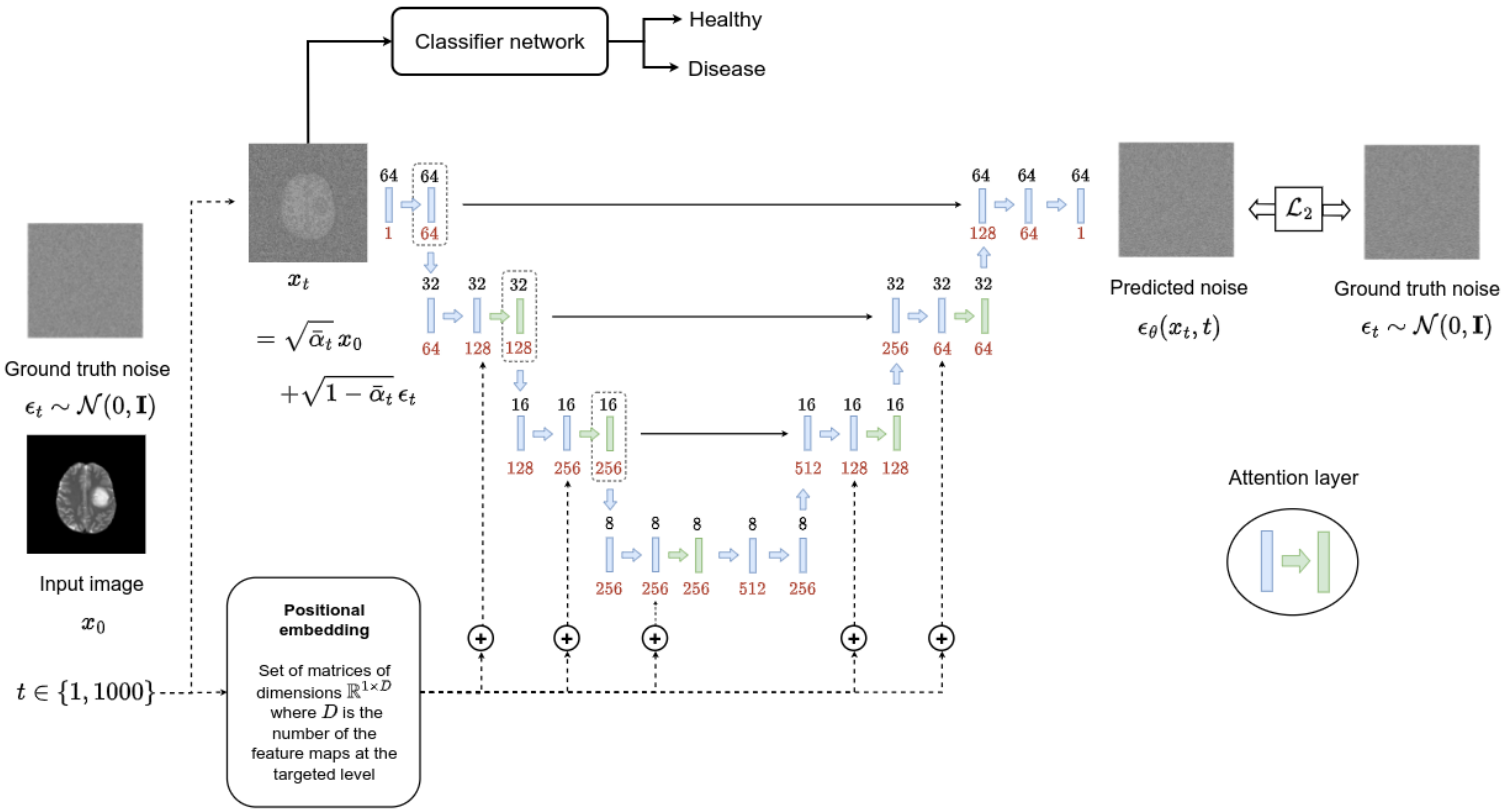
  - Make the diffusion process deterministic



Diffusion models for anomaly detection

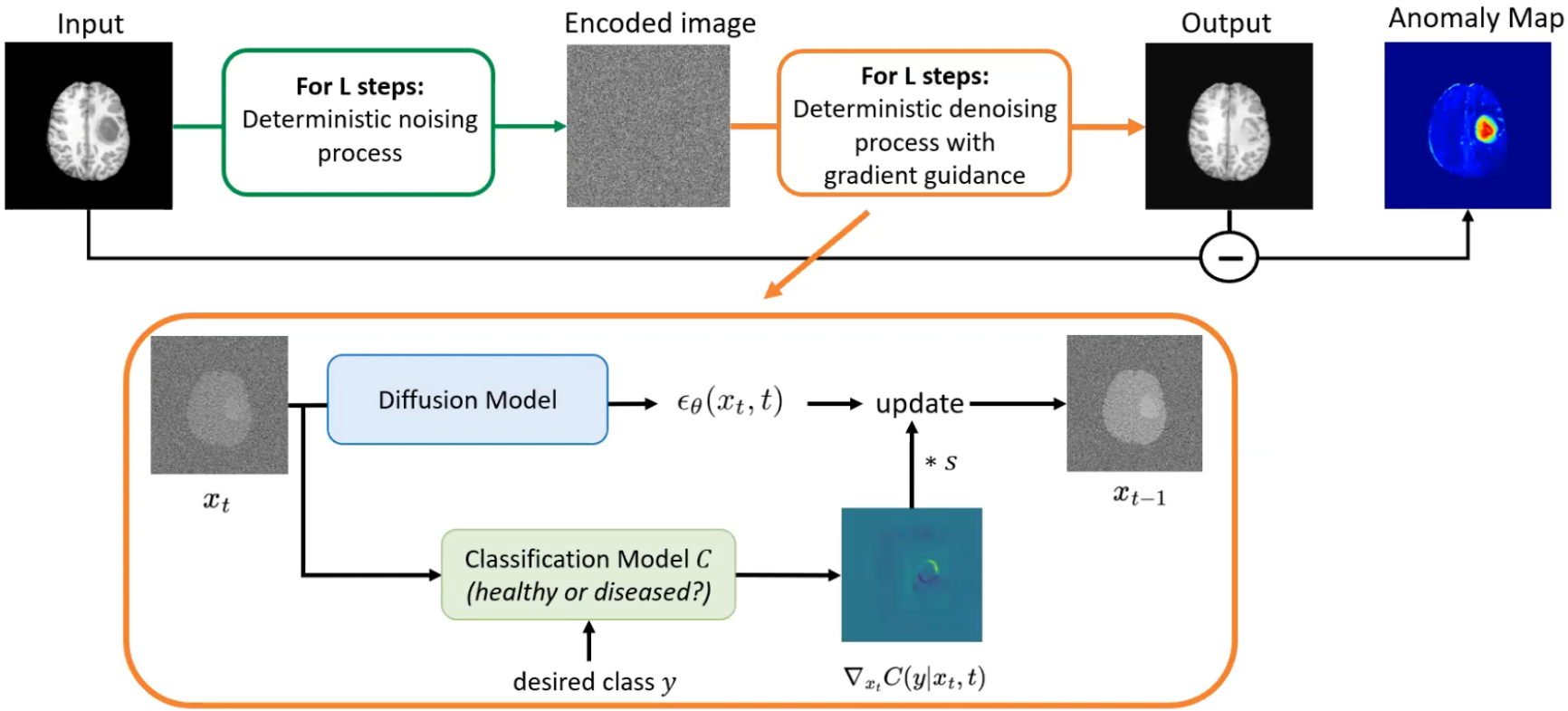
- Main algorithm – part 1

  - Train a classical DDPM on the dataset containing healthy and disease images
  - Train a classifier network C to predict the class label (healthy vs disease) from any noisy images x_t



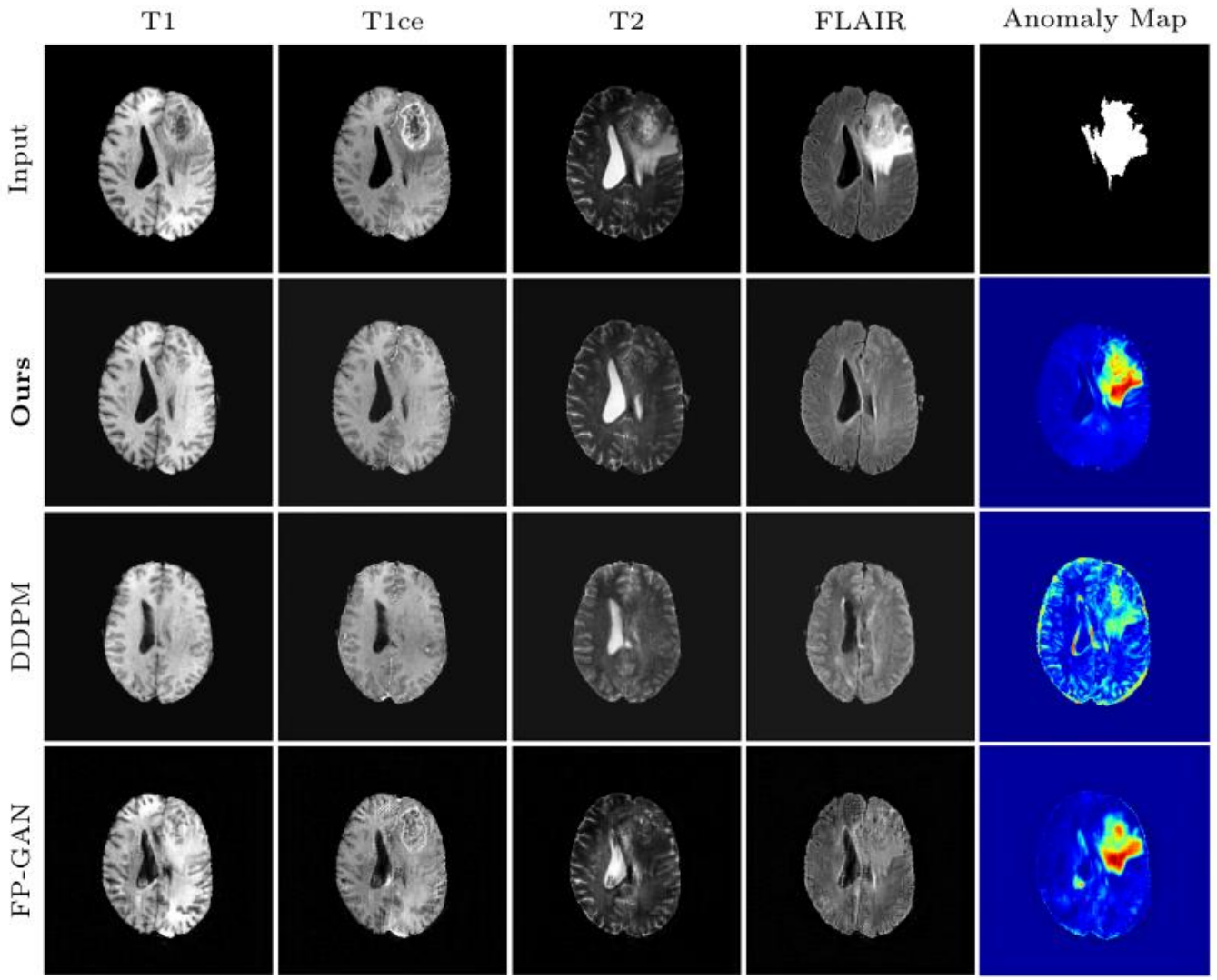
Diffusion models for anomaly detection

- Main algorithm – part 2
  - Use DDIM process
  - Compute the gradient of the classifier to guide the removing of anomaly regions



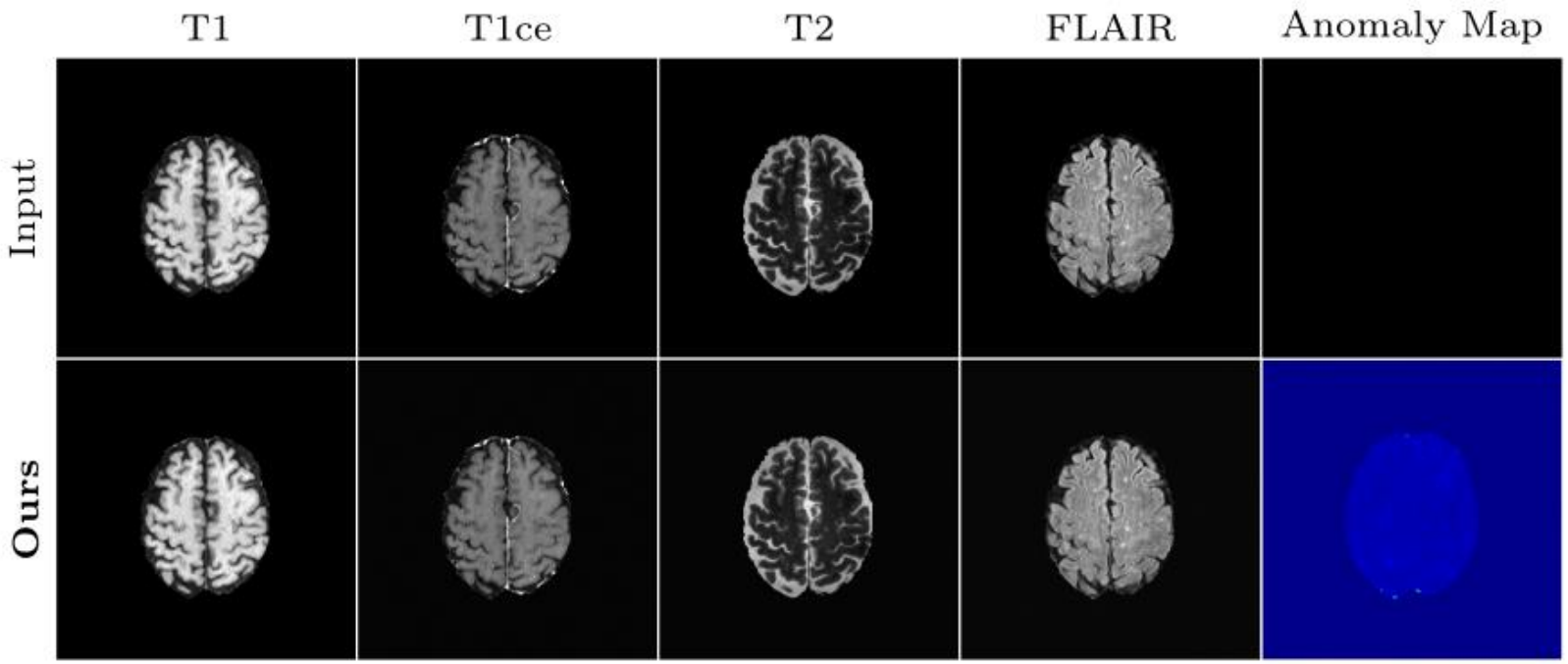
Diffusion models for anomaly detection

- Result on an image with a tumor



Diffusion models for anomaly detection

- Result on an image without any tumor

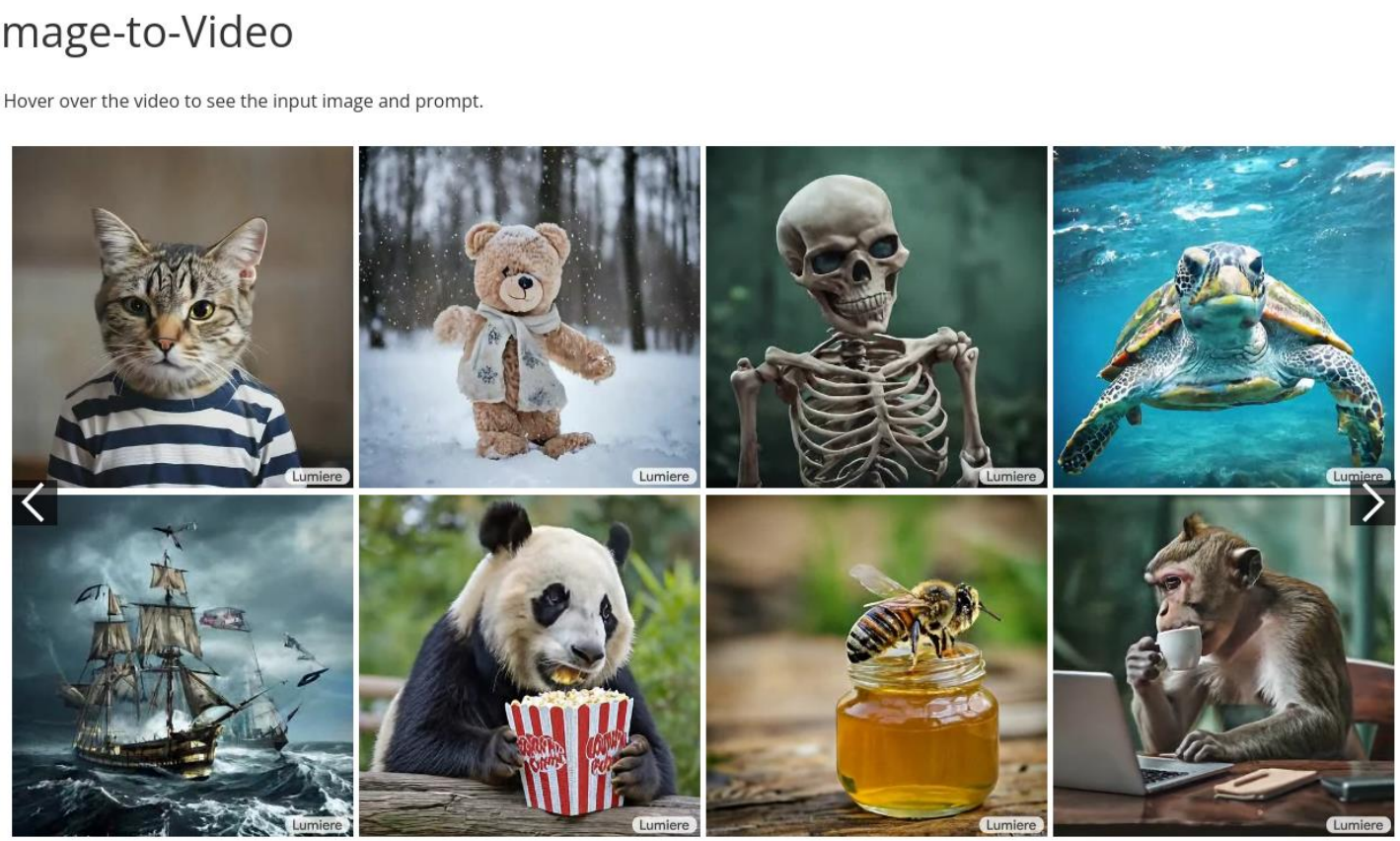


That’s all folks

What is the purpose of diffusion models?

- Recent extensions for video synthesis

https://lumiere-video.github.io/#section_image_to_video



Latent diffusion model (LDM)

- Random generation of synthetic images *with conditioning on text* learned from LAION-400M database

  ➔ Using the BERT tokenizer

  ➔ This model has over 1.45 billion parameters!

'A painting of the last supper by Picasso.'