CS480/680: Introduction to Machine Learning

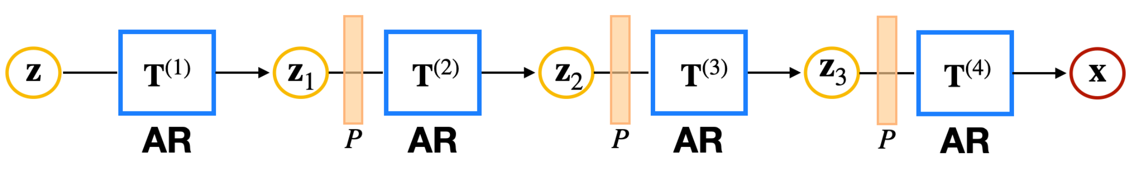
Lec 15: Diffusion Models

Yaoliang Yu

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS DAVID R. CHERITON SCHOOL OF COMPUTER SCIENCE

March 19, 2024

Auto-Regressive (AR) Flow Recalled



$$(\mathbf{T}_{\#}r)(\mathbf{x}) = r(\mathbf{z}) \;/\; \det\left(\nabla \mathbf{T}^{(1)}\mathbf{z}\right) \;/\; \det\left(\nabla \mathbf{T}^{(2)}\mathbf{z}_1\right) \;/\; \det\left(\nabla \mathbf{T}^{(3)}\mathbf{z}_2\right) \;/\; \det\left(\nabla \mathbf{T}^{(4)}\mathbf{z}_3\right)$$

$$x_j = z_j \cdot \exp\left(\alpha_j(z_1, \ldots, z_{j-1})\right) + \mu_j(z_1, \ldots, z_{j-1}) =: T_j(z_1, \ldots, z_{j-1}, z_j)$$

Now let the number of layers approach ∞!

Neural Ordinary Differential Equations (ODE)

$$\mathbf{x}_{t+1} \approx \mathbf{x}_t + \eta_t \cdot \mathbf{f}_t(\mathbf{x}_t) =: \mathbf{T}_t(\mathbf{x}_t)$$
$$\mathrm{d}\mathbf{x}_{t+1} = \mathbf{f}_t(\mathbf{x}_t)\,\mathrm{d}t$$

- Suppose $\mathbf{x}_t \sim p_t$
- Apply change-of-variable-formula we know $\mathbf{x}_{t+1} \sim p_{t+1}$, where

$$\begin{aligned}\log p_{t+1}(\mathbf{x}_{t+1}) &= \log p_t(\mathbf{x}_t) - \log|\det \partial_{\mathbf{x}} \mathbf{T}_t(\mathbf{x}_t)| \\ &= \log p_t(\mathbf{x}_t) - \log|\det[\mathrm{Id} + \eta_t \cdot \partial_{\mathbf{x}} \mathbf{f}_t(\mathbf{x}_t)]| \\ &\approx \log p_t(\mathbf{x}_t) - \eta_t \cdot \langle \partial_{\mathbf{x}}, \mathbf{f}_t(\mathbf{x}_t)\rangle\end{aligned}$$

- Continuous change-of-variable formula:

$$\boxed{\frac{\mathrm{d}\log p_t(\mathbf{x}_t)}{\mathrm{d}t} = -\langle \partial_{\mathbf{x}}, \mathbf{f}_t(\mathbf{x}_t)\rangle}$$

T. Q. Chen et al. "Neural ordinary differential equations". In: *Advances in Neural Information Processing Systems*. 2018, pp. 6572–6583.

Stochastic Differential Equations (SDE)

$$\mathrm{d}\mathbf{x}_{t+1} = \mathbf{f}_t(\mathbf{x}_t)\,\mathrm{d}t + G_t(\mathbf{x}_t)\,\mathrm{d}\mathbf{n}_t$$

$$\mathbf{x}_{t+1} \approx \mathbf{x}_t + \eta_t \cdot \mathbf{f}_t(\mathbf{x}_t) + \mathbf{g}_t(\mathbf{x}_t), \quad \text{where} \quad \mathbf{g}_t(\mathbf{x}_t) \sim \mathcal{N}(\mathbf{0}, \eta_t^2 G_t(\mathbf{x}_t) G_t(\mathbf{x}_t)^\top)$$

- $\mathbf{x}_{t+1}$ is now a noisy version of $\mathbf{x}_t$
- Suppose $\mathbf{x}_t \sim p_t$
- Kolmogorov forward equation (a.k.a. Fokker-Planck equation):

$$\boxed{\partial_t p_t = -\langle \partial_\mathbf{x}, p_t \mathbf{f}_t \rangle + \tfrac{1}{2} \left\langle \partial_\mathbf{x} \partial_\mathbf{x}^\top, p_t G_t G_t^\top \right\rangle}$$

- Kolmogorov backward equation (with fixed end time $t > s$):

$$-\partial_s p_s = \langle \mathbf{f}_s, \partial_\mathbf{x} p_s \rangle + \tfrac{1}{2} \left\langle G_s G_s^\top, \partial_\mathbf{x} \partial_\mathbf{x}^\top p_s \right\rangle$$

ODE $\Leftrightarrow$ SDE

$$
\begin{aligned}
\mathrm{d}\mathbf{x}_{t+1} &= \mathbf{f}_t(\mathbf{x}_t)\,\mathrm{d}t \\
\mathrm{d}\mathbf{x}_{t+1} &= \mathbf{f}_t(\mathbf{x}_t)\,\mathrm{d}t + G_t(\mathbf{x}_t)\,\mathrm{d}\mathbf{n}_t
\end{aligned}
$$

- Any ODE is a (trivial) SDE with $G_t \equiv \mathbf{0}$
- Conversely, any SDE is equivalent to an ODE:

$$
\boxed{\mathbf{f}_t \leftarrow \mathbf{f}_t - \tfrac{1}{2}(G_t G_t^\top)\partial_{\mathbf{x}} - \tfrac{1}{2}(G_t G_t^\top)\partial_{\mathbf{x}} \log p_t}
$$

- The score function plays an important role:

$$
\boxed{\mathbf{s}(\mathbf{x}) = \mathbf{s}_p(\mathbf{x}) := \partial_{\mathbf{x}} \log p(\mathbf{x})}
$$

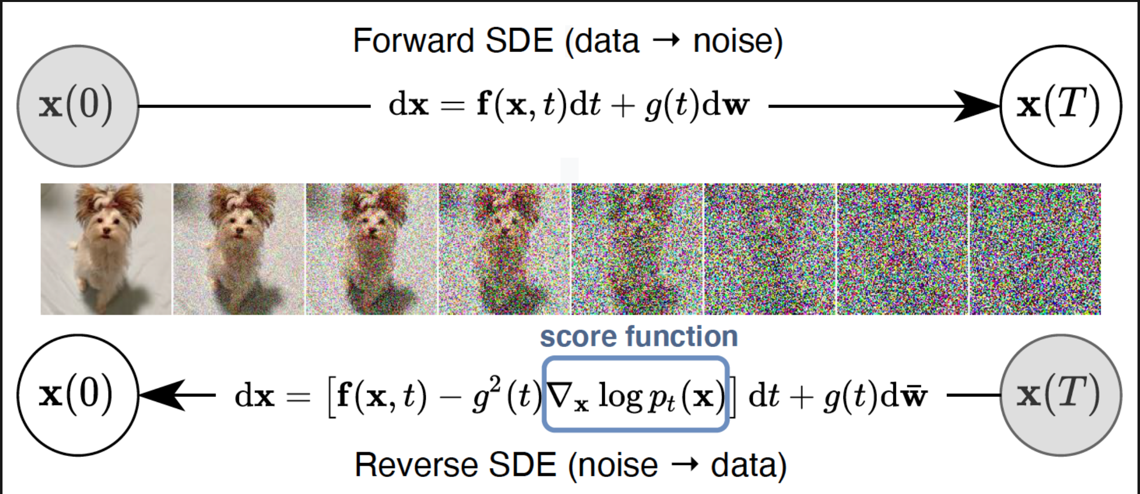
Reverse-time SDE

$$\mathrm{d}\mathbf{x}_{t+1} = \mathbf{f}_t(\mathbf{x}_t)\,\mathrm{d}t + G_t(\mathbf{x}_t)\,\mathrm{d}\mathbf{n}_t$$

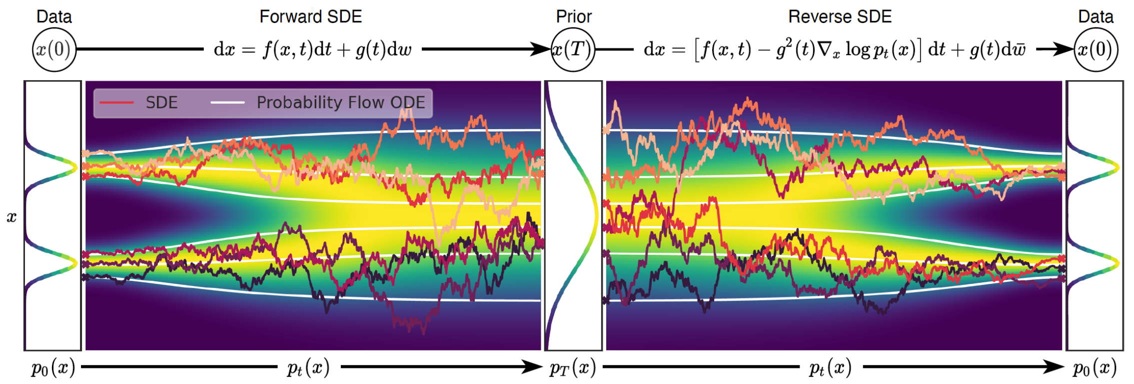
$$\mathrm{d}\bar{\mathbf{x}}_{t+1} = \bar{\mathbf{f}}_t(\bar{\mathbf{x}}_t)\,\mathrm{d}t + G_t(\bar{\mathbf{x}}_t)\,\mathrm{d}\bar{\mathbf{n}}_t, \quad \text{where} \quad \boxed{\bar{\mathbf{f}}_t = -\mathbf{f}_t + (G_t G_t^\top)\partial_{\mathbf{x}} + (G_t G_t^\top)\partial_{\mathbf{x}} \log p_t}$$

- Time flows backwards for the bar $\overline{\text{quantities}}$
- Forward SDE: diffuses date into noise
- Reverse SDE: molds noise into data
- $\mathbf{f}_t$ and G_t together specify $\bar{\mathbf{f}}_t$: key is to estimate the score $\partial_{\mathbf{x}} \log p_t$

B. D. O. Anderson. "Reverse-time diffusion equation models". *Stochastic Processes and their Applications*, vol. 12, no. 3 (1982), pp. 313–326.



Y. Song et al. "Score-Based Generative Modeling through Stochastic Differential Equations". In: *International Conference on Learning Representations*. 2021.



Y. Song et al. "Score-Based Generative Modeling through Stochastic Differential Equations". In: *International Conference on Learning Representations*. 2021.

Score Matching

$$
\begin{aligned}
\mathbb{F}(p\|q) &:= \tfrac{1}{2}\mathbb{E}_{\mathsf{X}\sim q}\|\partial_{\mathbf{x}}\log p(\mathsf{X}) - \partial_{\mathbf{x}}\log q(\mathsf{X})\|_2^2 \\
&= \mathbb{E}_{\mathsf{X}\sim q}\left[\tfrac{1}{2}\|\mathbf{s}_p(\mathsf{X})\|_2^2 + \langle \partial_{\mathbf{x}}, \mathbf{s}_p(\mathsf{X})\rangle + \tfrac{1}{2}\|\mathbf{s}_q(\mathsf{X})\|_2^2\right] \\
&\approx \hat{\mathbb{E}}_{\mathsf{X}\sim q}\left[\tfrac{1}{2}\|\mathbf{s}_p(\mathsf{X})\|_2^2 + \langle \partial_{\mathbf{x}}, \mathbf{s}_p(\mathsf{X})\rangle\right]
\end{aligned}
$$

- Under mild conditions, $\mathbb{F}(p\|q) = 0 \iff p \propto q$
- A Convenient way to estimate the score s_q and hence the density q
- The model score function s_p can be chosen as a neural net

A. Hyvärinen. "Estimation of Non-Normalized Statistical Models by Score Matching". *Journal of Machine Learning Research*, vol. 6, no. 24 (2005), pp. 695–709.

Denoising Auto-Encoder

- Suppose also have a latent variable Z with joint density $q(\mathbf{x}, \mathbf{z})$
- Exchange differentiation with integration we obtain:

$$\begin{aligned}
\mathbb{F}(p\|q) &:= \tfrac{1}{2}\mathbb{E}_{\mathsf{X}\sim q}\|\partial_{\mathbf{x}} \log p(\mathsf{X}) - \partial_{\mathbf{x}} \log q(\mathsf{X})\|_2^2 \\
&= \tfrac{1}{2}\mathbb{E}_{(\mathsf{X},\mathsf{Z})\sim q}[\|\mathbf{s}_p(\mathsf{X}) - \partial_{\mathbf{x}} \log q(\mathsf{X}|\mathsf{Z})\|_2^2 + \|\mathbf{s}_q(\mathsf{X})\|_2^2 - \|\partial_{\mathbf{x}} \log q(\mathsf{X}|\mathsf{Z})\|_2^2] \\
&\approx \tfrac{1}{2}\hat{\mathbb{E}}_{(\mathsf{X},\mathsf{Z})\sim q}\|\mathbf{s}_p(\mathsf{X}) - \partial_{\mathbf{x}} \log q(\mathsf{X}|\mathsf{Z})\|_2^2
\end{aligned}$$

- Useful when the conditional density $\partial_{\mathbf{x}} \log q(\mathsf{X}|\mathsf{Z})$ is easy to obtain

P. Vincent. "A Connection Between Score Matching and Denoising Autoencoders". *Neural Computation*, vol. 23, no. 7 (2011), pp. 1661–1674.

Score-based Diffusion Generative Models

$$
\begin{aligned}
\mathrm{d}\mathbf{x}_{t+1} &= \mathbf{f}_t(\mathbf{x}_t)\,\mathrm{d}t + G_t(\mathbf{x}_t)\,\mathrm{d}\mathbf{n}_t \\
\mathbf{x}_{t+1} &\approx \mathbf{x}_t + \eta_t \cdot \mathbf{f}_t(\mathbf{x}_t) + \mathbf{g}_t(\mathbf{x}_t), \quad \text{where} \quad \mathbf{g}_t(\mathbf{x}_t) \sim \mathcal{N}(\mathbf{0}, \eta_t^2 G_t(\mathbf{x}_t) G_t(\mathbf{x}_t)^\top)
\end{aligned}
$$

- Key is to estimate the score $\mathbf{s}_t(\mathbf{x}) = \partial_{\mathbf{x}} \log p_t$
- Apply denoising auto-encoder score matching:

$$
\min_{\boldsymbol{\theta}} \quad \hat{\mathbb{E}}_{t\sim\mu,(\mathsf{X}_t,\mathsf{X}_0)\sim q(\mathbf{x}_t,\mathbf{x}_0)} \lambda_t \| \mathbf{s}_t(\mathsf{X}_t;\boldsymbol{\theta}) - \partial_{\mathbf{x}} \log q(\mathsf{X}_t|\mathsf{X}_0) \|_2^2
$$

  - $\mathsf{X}_0 \sim q(\mathbf{x})$, the data density
  - $q(\mathbf{x}_t|\mathbf{x}_0)$ can be derived from the forward SDE, in closed-form if $\mathbf{f}_t$ is affine

Inference After Learning

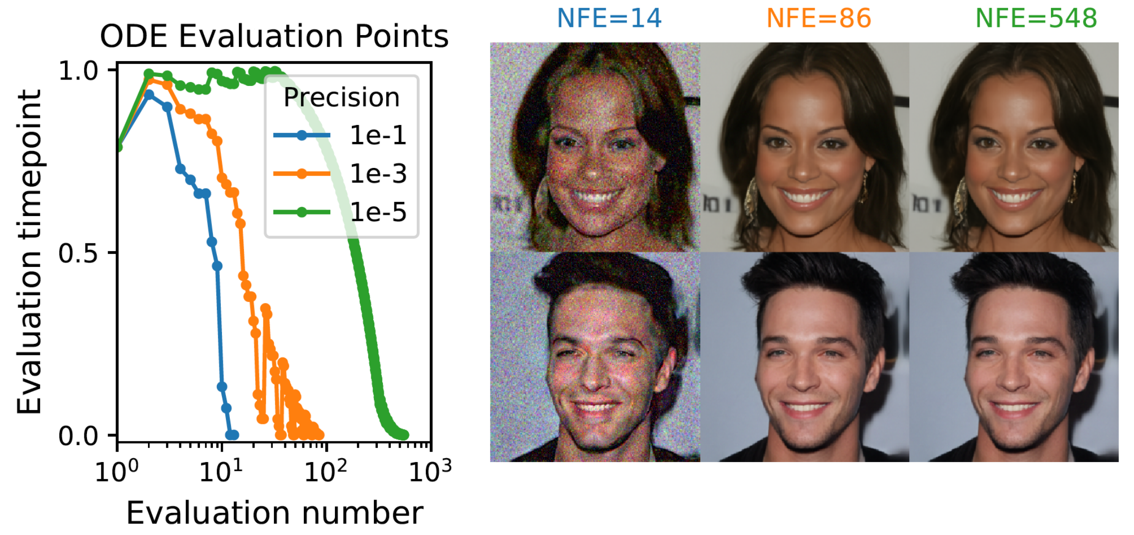
$$\mathrm{d}\mathbf{x}_{t+1} = \mathbf{f}_t(\mathbf{x}_t)\,\mathrm{d}t + G_t(\mathbf{x}_t)\,\mathrm{d}\mathbf{n}_t$$

$$\mathrm{d}\bar{\mathbf{x}}_{t+1} = \boxed{-\mathbf{f}_t + (G_t G_t^\top)\partial_{\mathbf{x}} + (G_t G_t^\top)\mathbf{s}_t(\bar{\mathbf{x}}_t;\boldsymbol{\theta})}\,\mathrm{d}t + G_t(\bar{\mathbf{x}}_t)\,\mathrm{d}\bar{\mathbf{n}}_t$$

$$\mathrm{d}\mathbf{x}_{t+1} = \boxed{\mathbf{f}_t - \tfrac{1}{2}(G_t G_t^\top)\partial_{\mathbf{x}} - \tfrac{1}{2}(G_t G_t^\top)\mathbf{s}_t(\mathbf{x}_t;\boldsymbol{\theta})}\,\mathrm{d}t$$

- Run the reverse SDE or the equivalent ODE
  - sample $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathrm{Id})$
  - apply numerical SDE or ODE solver (e.g. Euler-Maruyama)

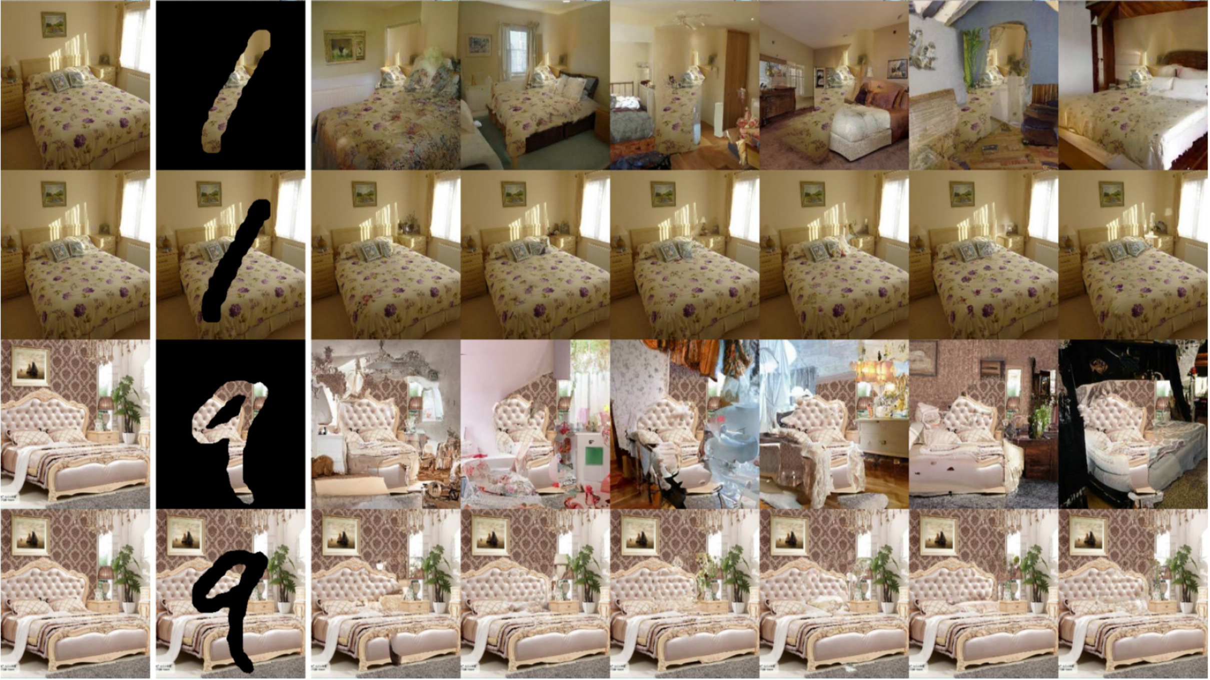
D. J. Higham. "An Algorithmic Introduction to Numerical Simulation of Stochastic Differential Equations". *SIAM Review*, vol. 43, no. 3 (2001), pp. 525–546.

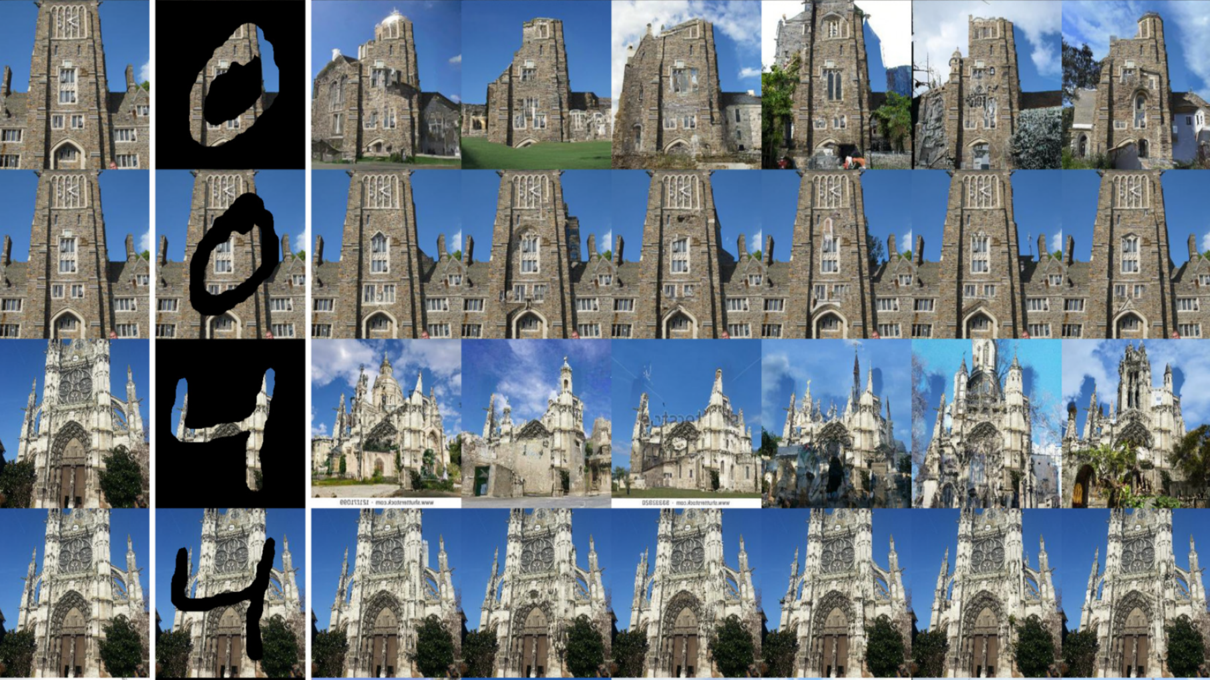


Y. Song et al. "Score-Based Generative Modeling through Stochastic Differential Equations". In: *International Conference on Learning Representations*. 2021.

Interpolation

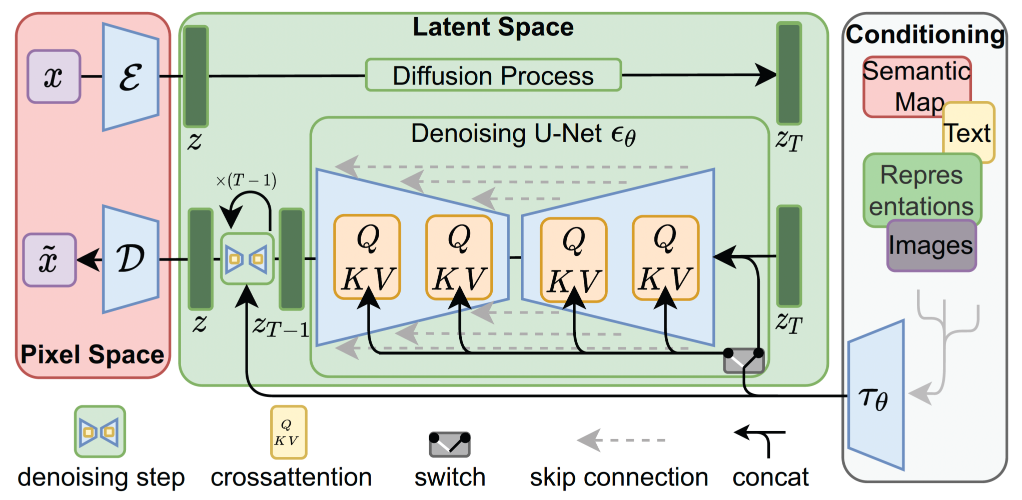
Y. Song et al. "Score-Based Generative Modeling through Stochastic Differential Equations". In: *International Conference on Learning Representations*. 2021.






shutterstock
www.shutterstock.com · 31170427

Stable Diffusion



R. Rombach et al. "High-Resolution Image Synthesis with Latent Diffusion Models". In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 10674–10685.