Simon Welker[*,†], Henry N. Chapman[†], Timo Gerkmann[*]

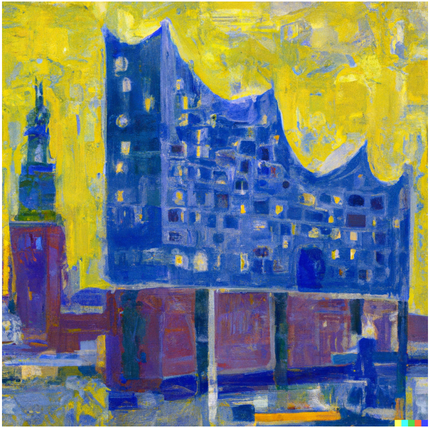
Blind Drifting: Diffusion models with a linear SDE drift term for blind image restoration tasks

The Symbiosis of Deep Learning and Differential Equations II

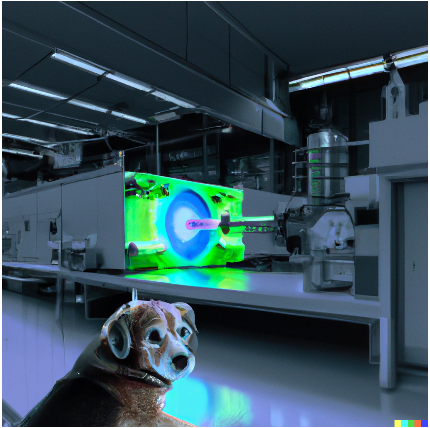
[*]Signal Processing, Department of Informatics, University of Hamburg
[†]Coherent Imaging Division, Center for Free-Electron Lasers (CFEL), DESY

December 9, 2022

Score-based Generative Models: DALL·E 2



"Elbphilharmonie, Hamburg, by Vincent van Gogh"



"Dog who has no idea what he's doing at an international Free Electron Laser research facility"

Score-based Generative Models (SGMs)

Score-based generative models (SGMs) consist of two processes:[1,2]

- *Forward process*, gradually adds noise to the input
- *Reverse process*, generates data by gradual denoising

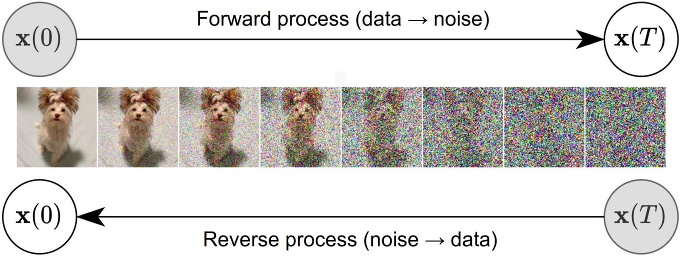


Figure: Graphic by Song et al. [2]

[1] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International Conference on Machine Learning*, 2015, pp. 2256–2265.

[2] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," *Int. Conf. on Learning Representations (ICLR)*, 2021.

- *Song et al.*[2]: Model forward process as a stochastic differential equation (SDE)

$$\mathrm{d}\mathbf{x} = \underbrace{\mathbf{f}(\mathbf{x}, t)}_{\text{Drift}}\,\mathrm{d}t + \underbrace{g(\mathbf{x}, t)}_{\text{Diffusion}}\,\mathrm{d}\mathbf{w}, \qquad t \in [0, 1] \tag{1}$$

which has an associated *Reverse SDE*:

$$\mathrm{d}\mathbf{x} = \left[-\mathbf{f}(\mathbf{x}, t) + g(\mathbf{x}, t)^2 \underbrace{\nabla_{\mathbf{x}} \log p(\mathbf{x})}_{\text{Score of } p(\mathbf{x})} \right] \mathrm{d}t + g(\mathbf{x}, t)\mathrm{d}\overline{\mathbf{w}} \tag{2}$$

- Learn *score network* $\mathbf{s}_\theta(\mathbf{x}, t)$ $\approx$ *score* $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$, plug $\mathbf{s}_\theta$ into (2) to create a **Reverse Neural SDE**
  - **As initial value**: $t = 1$, $x_{t=1}$ = pure Gaussian noise

➡ Numerically solving this initial value problem **generates** new samples!

[2] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," *Int. Conf. on Learning Representations (ICLR)*, 2021.

Score-based generative models (SGMs) for inverse problems?

- **Motivation**: SGMs exhibit strong modeling of data distributions
  $\rightarrow$ make inverse problems easier
- Some inverse problems have been tackled[3,4], but approaches typically assume **known and/or linear** forward operators T
- No such luxuries for blind inverse problems:
  - Blind deconvolution
  - Non-uniform deblurring
  - Artifact removal
  - Background removal
  - ...

[3] B. Kawar, M. Elad, S. Ermon, and J. Song, "Denoising diffusion restoration models," in *ICLR Workshop on Deep Generative Models for Highly Structured Data*, 2022. [Online]. Available: https://openreview.net/forum?id=BExXihVOvWq.

[4] H. Chung, J. Kim, M. T. Mccann, M. L. Klasky, and J. C. Ye, "Diffusion posterior sampling for general noisy inverse problems," *arXiv preprint arXiv:2209.14687*, 2022.

SDEs of classic SGMs

- Many SGMs use something like the Variance Exploding SDE:

$$\mathbf{f}(\mathbf{x}, t) = 0\,, \qquad g(t) = \left[\sigma_{\text{min}} \left(\frac{\sigma_{\text{max}}}{\sigma_{\text{min}}}\right)^t\right] \tag{3}$$

- Much exploration of the diffusion term g in the literature
- But very little of the **drift** $\mathbf{f}$
- Our idea: **Utilize the drift term** for our blind problems

SDEs for unknown forward operators (1)

- Let ($\mathbf{x}_0$ = clean image, $\mathbf{y}$ = corrupted image) **from a paired dataset**
- **Idea:** Realize increasing corruption of image via new forward SDE

Variance Exploding SDE

$$\mathrm{d}\mathbf{x}_t = \underbrace{\left[\sigma_{\text{min}} \left(\frac{\sigma_{\text{max}}}{\sigma_{\text{min}}}\right)^t\right]}_{\text{Diffusion } g} \mathrm{d}\mathbf{w}$$

Our proposal: Add a drift term

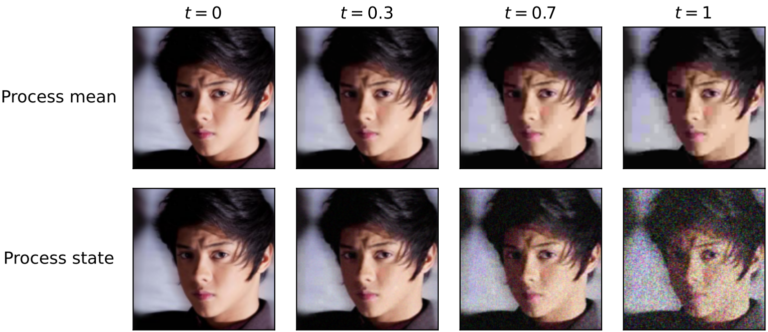
$$\mathrm{d}\mathbf{x}_t = \underbrace{\gamma F(t)(\mathbf{y} - \mathbf{x}_t)}_{\text{Drift } \mathbf{f}}\mathrm{d}t + \underbrace{\left[\sigma_{\text{min}} \left(\frac{\sigma_{\text{max}}}{\sigma_{\text{min}}}\right)^t\right]}_{\text{Diffusion } g} \mathrm{d}\mathbf{w}$$

- Drift $\mathbf{f} = \gamma F(t)(\mathbf{y} - \mathbf{x}_t)$ pulls $\mathbf{x}_t$ towards $\mathbf{y}$
  - ➡ adds the corruption in $\mathbf{y}$ over increasing process time t
  - ➡ $\mathbf{f}$ affine in $\mathbf{x}_t \Rightarrow$ closed-form mean and variance

➡ **Solving the Reverse SDE removes the corruption**

SDEs for unknown forward operators (2)

Forward process, illustrated



- Adds relatively little noise even at $t = 1.0$

➡ Diffuses within a much smaller region than classic SGMs

Experiments

- Task: JPEG Artifact Removal
  - Example task with nonlinear forward operator
  - We **do not** inform models about quality factor or how JPEG works

- Train NCSN++ network[2] on *CelebA-HQ 256x256*
  - Random JPEG quality factors 0–30%
  - **Proposed:** Train as task-adapted score model
  - **Baseline:** Train via simple L_2 regression

- Forward SDE drift term: $\mathbf{f} = \gamma(\mathbf{y} - \mathbf{x}_t)$
  - ➡ {Ornstein-Uhlenbeck + Variance Exploding} (OUVE) SDE
- Solve Reverse SDE with Euler-Maruyama, $N = 100$ steps

[2] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," *Int. Conf. on Learning Representations (ICLR)*, 2021.

JPEG Artifact Removal: Results (1)

- Input JPEG quality level 10%:

✔ Outperforms regression baseline with same network architecture

✔ Much better at fine details & modeling the clean image distribution

JPEG Artifact Removal: Results (2)

	KID↓	FID↓	LPIPS↓	SSIM↑
Corrupted	22.53	36.26	0.20	0.82
Baseline	38.18	45.92	0.13	**0.90**
Ours (OUVE)	**2.37**	**15.69**	**0.08**	0.83

Table: Metrics for JPEG quality factor 10%

✔ Our SGM beats the L_2 baseline perceptually and in terms of distribution metrics

Conclusion

- Power of the SDE formalism for SGMs should be explored further
  - Drift term $\mathbf{f}$ is often ignored

- A simple affine drift term can **adapt SGMs to image-to-image tasks**
  - Qualitatively different outputs than regression baseline even with
    - the same training dataset
    - the same DNN architecture
    - no loss engineering or feature engineering
  - ✔ Training: similarly simple as supervised regression
  - ✔ Needs relatively few steps even with basic solvers
  - ✔ No intricate knowledge of corruption needed

References

[1] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International Conference on Machine Learning*, 2015, pp. 2256–2265.

[2] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," *Int. Conf. on Learning Representations (ICLR)*, 2021.

[3] B. Kawar, M. Elad, S. Ermon, and J. Song, "Denoising diffusion restoration models," in *ICLR Workshop on Deep Generative Models for Highly Structured Data*, 2022. [Online]. Available: https://openreview.net/forum?id=BExXihVOvWq.

[4] H. Chung, J. Kim, M. T. Mccann, M. L. Klasky, and J. C. Ye, "Diffusion posterior sampling for general noisy inverse problems," *arXiv preprint arXiv:2209.14687*, 2022.

[5] M. Ehrlich, L. Davis, S.-N. Lim, and A. Shrivastava, "Quantization guided JPEG artifact correction," in *Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 293–309.

Denoising Score Matching

Training objective: Minimize L^2 loss between model output and the score

$$\|\mathbf{s}_\theta(\mathbf{x}, \mathbf{y}, t) - \nabla_{\mathbf{x}} \log p_{0t}(\mathbf{x}|\mathbf{x}_0, \mathbf{y})\|_2^2 . \tag{4}$$

When p_{0t} is Gaussian, its score equals

$$\nabla_{\mathbf{x}} \log p_{0t}(\mathbf{x}|\mathbf{x}_0, \mathbf{y}) = -\frac{\mathbf{x} - \boldsymbol{\mu}(\mathbf{x}_0, \mathbf{y}, t)}{\sigma(t)^2} = \frac{-\mathbf{z}}{\sigma(t)} , \tag{5}$$

where $\mathbf{x} = \boldsymbol{\mu}(\mathbf{x}_0, \mathbf{y}, t) + \sigma(t)\mathbf{z}$, $\mathbf{z} \sim \mathcal{N}_{\mathbb{C}}(0, \mathbf{I})$.

➡ $\mathbf{s}_\theta$ is trained as a task-adapted Gaussian denoiser, able to deal with multiple noise scales

Additional results

- QGAC[5] is a SotA method for JPEG artifact removal
- TSDVE: $\mathbf{f} = \gamma t(\mathbf{y} - \mathbf{x}_t)$

	KID↓	FID↓	LPIPS↓	SSIM↑
Corrupted	22.53	36.26	0.20	0.82
QGAC	46.89	53.97	0.14	**0.90**
Baseline	38.18	45.92	0.13	**0.90**
Ours (TSDVE)	**2.32**	15.72	**0.08**	0.83
Ours (OUVE)	2.37	**15.69**	**0.08**	0.83

Table: Metrics for JPEG quality factor 10%

[5] M. Ehrlich, L. Davis, S.-N. Lim, and A. Shrivastava, "Quantization guided JPEG artifact correction," in *Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 293–309.