Image Generation with Diffusion Models

Forward Diffusion (Noising with White Gaussian Noise)

x_0 $\quad x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\, \varepsilon$ with $\varepsilon \sim \mathcal{N}(0_d, \mathrm{I}_{d\times d})$ $\quad x_{1000}$

$\hat{x}_0$ $\quad \hat{x}_t$ $\quad \hat{x}_{1000}$

Reverse Diffusion (Generation using the trained Denoising Auto-Encoder)

Image from [2]

Examples of Diffusion models:
- **Stable Diffusion** (SD) [1]
- DALL-E
- Imagen
- Sora
- Stable Video Diffusion

Diffusion in Style

Martin Nicolas Everaert
Marco Bocchio
Sami Arpa
Sabine Süsstrunk
Radhakrishna Achanta

ICCV23 PARIS

The initial latent tensor $\hat{x}_{1000}$ affects the composition and style of the generated image $\hat{x}_0$, so adapting it to the style facilitates style adaptation.

We fine-tune Stable Diffusion (SD) [1] with a **style-specific noise distribution** $\mathcal{N}(\mu_{style}, \Sigma_{style})$ instead of the default $\mathcal{N}(0_d, \mathrm{I}_{d\times d})$.

Original diffusion $\varepsilon \sim \mathcal{N}(0_d, \mathrm{I}_{d\times d})$

Our style-adapted diffusion $\varepsilon \sim \mathcal{N}(\mu_{style}, \Sigma_{style})$

x_0 $\quad x_{333}$ $\quad x_{667}$ $\quad x_{1000}$

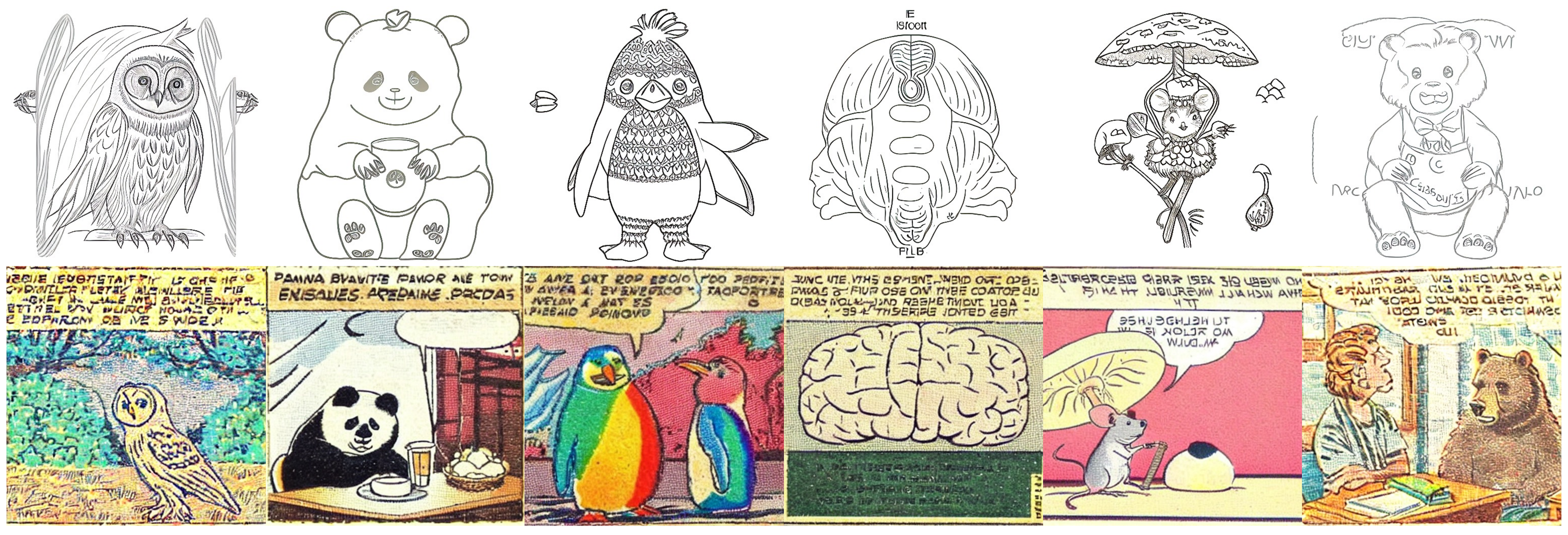
We compute the style-specific noise parameters μ_{style} and Σ_{style} from **a small set of images of the desired style**.

Apart from the style-specific noise distribution $\mathcal{N}(\mu_{style}, \Sigma_{style})$, the fine-tuned model **can be used like the original model**.

We use our approach to fine-tune SD 1.5 [1] to different styles, *e.g.* anime sketches, or comics images.

Style 1, anime sketches [3]

Style 2, comics images [4]



A side view of an owl sitting in a field. | *A panda making latte art.* | *Rainbow coloured penguin.* | *A cross-section view of a brain.* | *A mouse using a mushroom as an umbrella.* | *A confused grizzly bear in calculus class.*

We sample the initial latent tensor $\hat{x}_{1000}$ from the style-specific noise distribution and use the fine-tuned model to iteratively denoise it.

$\hat{x}_{1000}$ $\quad \hat{x}_{667}$ $\quad \hat{x}_{333}$ $\quad \hat{x}_0$ $\quad \hat{x}_{1000}$ $\quad \hat{x}_{667}$ $\quad \hat{x}_{333}$ $\quad \hat{x}_0$

Exploiting the Signal-Leak Bias in Diffusion Models

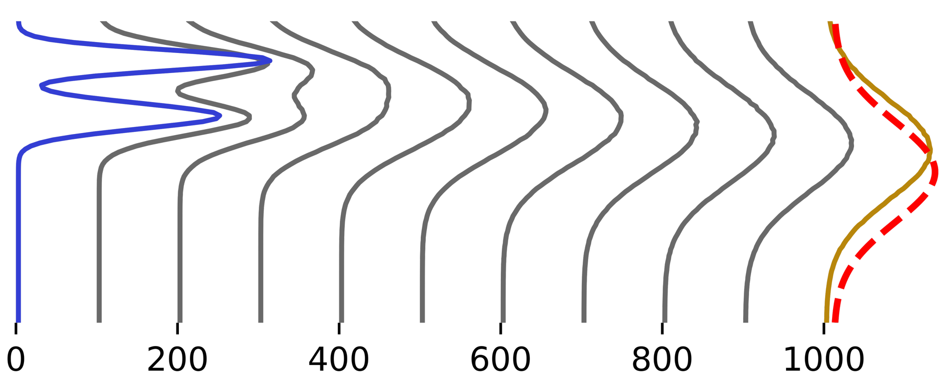
Martin Nicolas Everaert
Athanasios Fitsios
Marco Bocchio
Sami Arpa
Sabine Süsstrunk
Radhakrishna Achanta

JAN 4-8 WACV 2024 WAIKOLOA HAWAII

Common **diffusion models never fully corrupt images** during training [5,6]:

$$x_{1000} = \sqrt{\bar{\alpha}_{1000}}\, x_0 + \sqrt{1-\bar{\alpha}_{1000}}\, \varepsilon \text{ with } x_0 \sim p(x_0) \text{ and } \varepsilon \sim \mathcal{N}(0_d, \mathrm{I}_{d\times d})$$
$$\approx 0.068\, x_0 + 0.998\, \varepsilon$$

However, the process of **generating images starts with pure noise** $\hat{x}_{1000} \sim \mathcal{N}(0_d, \mathrm{I}_{d\times d})$, oblivious of the **signal leak** $\sqrt{\bar{\alpha}_{1000}}\, x_0$ present in x_{1000} during training, **creating a bias**.



$\hat{x}_{1000} \sim \mathcal{N}(0_d, \mathrm{I}_{d\times d}) \neq p(x_{1000})$

The diffusion model uses the signal-leak $\sqrt{\bar{\alpha}_T}\, x_0$ to deduce the **low-frequency information** about x_0 from x_{1000}.

Using $\hat{x}_{1000} \sim \mathcal{N}(0_d, \mathrm{I}_{d\times d})$ **biases** the low-frequency components towards **medium values**.

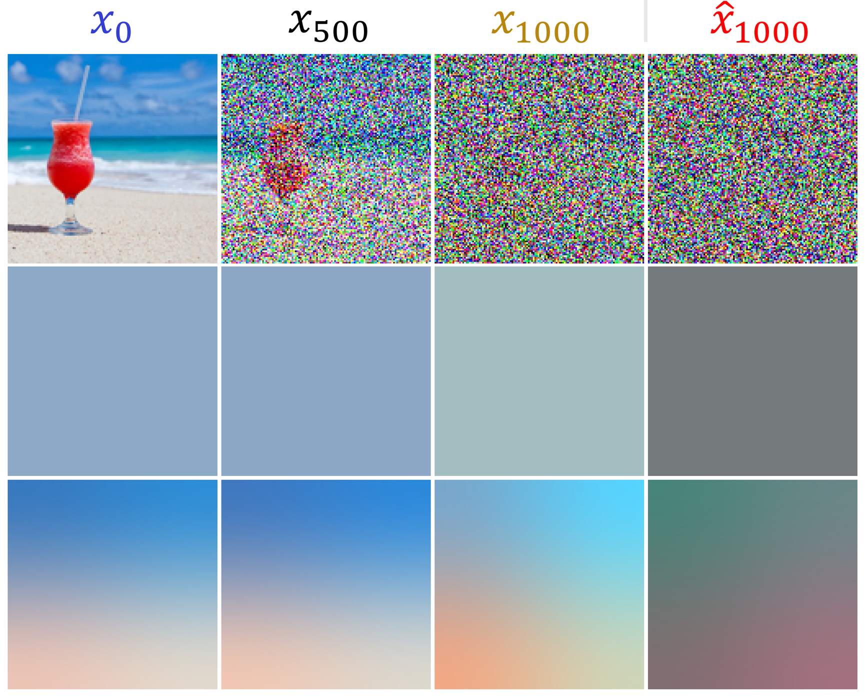
Training | Inference

x_0 $\quad x_{500}$ $\quad x_{1000}$ $\quad \hat{x}_{1000}$

x_t

Average value $(/\sqrt{\bar{\alpha}_t})$

3 lowest frequency components $(/\sqrt{\bar{\alpha}_t})$



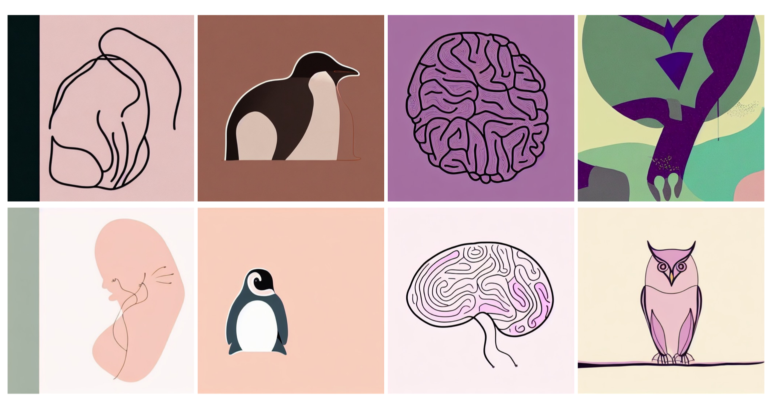
Instead of retraining or finetuning [5,6,A] to remove this bias, we exploit it to our advantage, generating images in the style we want, **include a signal-leak** $\sqrt{\bar{\alpha}_T}\, \tilde{x}$ in $\hat{x}_{1000}$ **at inference time**, starting generating images from:

$$\hat{x}_{1000} = \sqrt{\bar{\alpha}_{1000}}\, \tilde{x} + \sqrt{1-\bar{\alpha}_{1000}}\, \varepsilon \quad \text{with } \tilde{x} \sim q(\tilde{x}) \text{ and } \varepsilon \sim \mathcal{N}(0_d, \mathrm{I}_{d\times d})$$

With $q(\tilde{x}) = \mathcal{N}(\mu_{style}, \Sigma_{style})$, we exploit the bias to generate images $\hat{x}_0$ in the style we want:

"line-art" model [7]

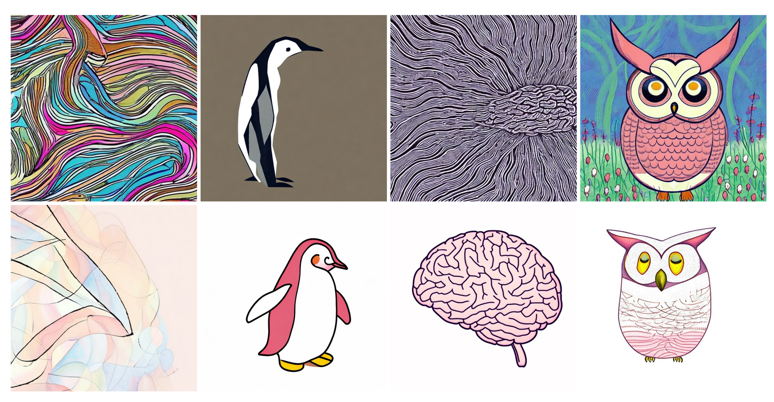
"line-art" model [7] **with ours**



SD 2.1 [1]

"[…] in the style of line art, pastel colors, white background."

SD 2.1 [1] **with ours**



"nasa space" model [8]

"nasa space" model [8] **with ours**

SD 2.1 [1]

"A blue city at night, long exposure, orange and blue."

SD 2.1 [1] **with ours**

By modeling separately the low-frequency components in the frequency domain, and setting them manually in $\tilde{x}$ **at inference time**, we can control the low-frequency components (here, the mean color) of the generated images $\hat{x}_0$:

References:

[A] Everaert et al. "Diffusion in style." *ICCV* 2023.
[B] Everaert et al. "Exploiting the signal-leak bias in diffusion models." *WACV* 2024.
[1] Rombach et al. "High-resolution image synthesis with latent diffusion models." *CVPR* 2022.
[2] Nichol and Dhariwal. "Improved denoising diffusion probabilistic models." *ICML* 2021.
[3] Taebum. "Anime Sketch Colorization dataset." Kaggle dataset. 2018.
[4] Simon and Kirby. "48 Famous Americans." 1947.
[5] Guttenberg. "Diffusion with Offset Noise." 2023
[6] Lin et al. "Common Diffusion Noise Schedules and Sample Steps are Flawed." *WACV* 2024.
[7] Karan. "line-art" model. via huggingface.co/sd-concepts-library. 2022.
[8] MatAlart. "nasa space" model. Via huggingface.co/sd-dreambooth-library. 2022

https://ivrl.github.io/diffusion-in-style/

https://ivrl.github.io/signal-leak-bias/

Acknowledgement:
The 2 works were supported by Innosuisse grant 48552.1 IP-ICT.

EPFL IVRL