Image generation with shortest path diffusion

Ayan Das [*] [1] **Stathi Fotiadis** [*] [1 2] **Anil Batra** [1 3] **Farhang Nabiei** [1] **FengTing Liao** [1] **Sattar Vakili** [1] **Da-shan Shiu** [1] **Alberto Bernacchia** [1]

Abstract

The field of image generation has made significant progress thanks to the introduction of Diffusion Models, which learn to progressively reverse a given image corruption. Recently, a few studies introduced alternative ways of corrupting images in Diffusion Models, with an emphasis on blurring. However, these studies are purely empirical and it remains unclear what is the optimal procedure for corrupting an image. In this work, we hypothesize that the optimal procedure minimizes the length of the path taken when corrupting an image towards a given final state. We propose the Fisher metric for the path length, measured in the space of probability distributions. We compute the shortest path according to this metric, and we show that it corresponds to a combination of image sharpening, rather than blurring, and noise deblurring. While the corruption was chosen arbitrarily in previous work, our Shortest Path Diffusion (SPD) determines uniquely the entire spatiotemporal structure of the corruption. We show that SPD improves on strong baselines without any hyperparameter tuning, and outperforms all previous Diffusion Models based on image blurring. Furthermore, any small deviation from the shortest path leads to worse performance, suggesting that SPD provides the optimal procedure to corrupt images. Our work sheds new light on observations made in recent works and provides a new approach to improve diffusion models on images and other types of data.

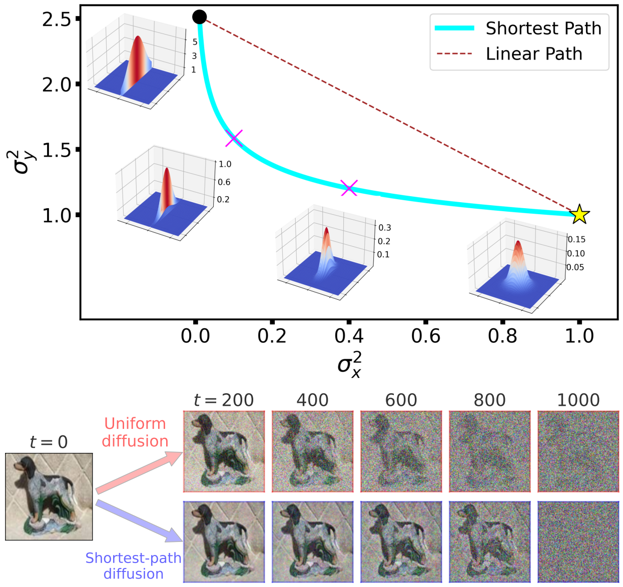

Figure 1. **Illustration of shortest path**. **Top**: Transformation of a bivariate Gaussian distribution, parameterized by the variance along two orthogonal directions, σ_x^2 and σ_y^2. The initial distribution has $\sigma_x^2 \ll \sigma_y^2$ (black circle), while the final distribution has $\sigma_x^2 = \sigma_y^2$ (yellow star, isotropic noise). According to the Fisher metric, the shortest path between the two distributions is *not* the linear path (dashed line), instead is given by the curved path (cyan curve), in which σ_y^2 decreases first, and σ_x^2 increases later. **Bottom**: Comparison of shortest path and uniform noising for image corruption. In uniform noising, the original image dissipates while uniform noise appears. Instead, the shortest path corresponds to image sharpening and noise deblurring. Lower frequencies of the image dissipate before higher frequencies. Similarly, noise appears at lower frequencies first and higher frequencies later. Figure 3 shows how signal and noise change in time for different frequencies.

1. Introduction

The field of image generation has seen rapid progress since the introduction of algorithms based on deep learning (Bond-Taylor et al., 2022). A common approach is using a deep neural network to map input noise into an output image, a fast process that requires a single forward pass. These methods include Generative Adversarial Networks (Goodfellow

[*] Equal contribution [1] MediaTek Research, Cambourne, UK [2] Department of Bioengineering, Imperial College London, London, UK [3] School of Informatics, University of Edinburgh, Edinburgh, UK. Correspondence to: Ayan Das <ayan.das@mtkresearch.com>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

et al., 2020; Karras et al., 2020), which provide good image quality, Variational Autoencoders (Kingma & Welling, 2014; Child, 2021) and Normalizing Flows (Dinh et al., 2017; Chen et al., 2019), which provide a rich diversity of sampled images. Other approaches based on deep learning include Autoregressive Models (Van Den Oord et al., 2016; Child et al., 2019), which generate one pixel (or patch) at a time.

Diffusion models (Sohl-Dickstein et al., 2015; Song & Ermon, 2019; Ho et al., 2020) represent an alternative family of algorithms outperforming previous approaches both in terms of quality (Dhariwal & Nichol, 2021) and diversity (Kingma et al., 2021). Similar to previous approaches, diffusion models transform input noise into an output image. However, instead of generating an image by a single forward pass through a neural network, diffusion models use multiple steps of denoising, which require multiple forward passes. This iterative procedure allows refining an image to unprecedented quality. The combination of diffusion and language models led to impressive progress in text-to-image generation (Saharia et al., 2022; Nichol et al., 2022).

Recent work questioned the procedure for corrupting images in diffusion models. Early work proposed using noise (Sohl-Dickstein et al., 2015; Song & Ermon, 2019; Ho et al., 2020), but recent studies explored alternative procedures, with a strong focus on image blurring (Rissanen et al., 2023; Lee et al., 2022a; Bansal et al., 2022; Daras et al., 2022; Hoogeboom & Salimans, 2023). In all previous work, corruptions are chosen arbitrarily and it remains unclear what is the optimal procedure for corruption. In this work, we provide a candidate for the optimal procedure. We reason that the main sources of errors in diffusion models are the approximations made in reversing the corruption. Therefore, the optimal corruption procedure would be one that minimizes those errors.

A procedure for corrupting images in a diffusion model is equivalent to a transformation of the data distribution into another probability distribution, which is often taken to be an isotropic Gaussian. For a given parameterization of the probability distributions, this transformation can be visualized by a path in the space of parameters, from the parameter values of the data distribution to those of an isotropic Gaussian (see figure 1). Any given procedure for corruption corresponds to a path in the space of distributions. We hypothesize that the optimal procedure for corruption corresponds to the shortest path. The intuition is that any error made in approximating the true reversal of the corruption, accumulated along the path, would be smaller if the path is shorter.

To compute the path length, a metric in the space of distributions needs to be defined. We choose the Fisher Information Matrix as metric, because it bounds the precision of maximum likelihood estimation (Amari, 2016), and Diffusion Models are trained by using a bound on the likelihood as the loss function (Sohl-Dickstein et al., 2015; Ho et al., 2020). Furthermore, the Fisher metric is reparameterization invariant, namely any length computed by the metric does not depend on the choice made for parameterizing the distributions (Amari, 2016). We contribute the following:

- We compute analytically the shortest path between Gaussian distributions, and we propose an approximation for the non-Gaussian case (e.g. image data).
- We show that the shortest path corresponds to a combination of image sharpening and noise deblurring. We provide the exact formula for the specific corruption prescribed by the shortest path.
- We test our Shortest Path Diffusion (SPD) on CIFAR10. We show that any small departure from the shortest path results in worse performance, and SPD outperforms all methods based on image blurring. Our results suggest that SPD provides the optimal corruption.
- We also test SPD on ImagNet 64×64, on the task of *unconditional* generation, and we show that SPD improves on strong baselines without any hyperparameter tuning.

2. Related work

2.1. Diffusion Models

Diffusion probabilistic models were introduced by Sohl-Dickstein et al. (2015) following a line of research on Markov chain-based generative models (Bengio et al., 2014; Salimans et al., 2015). The work of Song & Ermon (2019; 2020) proposed an algorithm for image generation based on learning the score of the data distribution. The work of Ho et al. (2020) pointed out the equivalence between diffusion and score-based generative models, and showed the capability of these models in generating high-quality images. The two approaches were unified by the framework of stochastic differential equations (Song et al., 2020). The work of Song et al. (2021) introduced non-Markovian diffusion models, allowing deterministic and faster sampling.

2.2. Variance schedule

One of the first avenues to improve the performance of Diffusion Models was adjusting the temporal properties of the corruption process, also known as variance schedule, while its spatial properties remained fixed to isotropic noise. Early work used a linear or exponential increase of variance. The work of Nichol & Dhariwal (2021a) introduced a cosine function of time, which enabled better quality of generated images. The work of Kingma et al. (2021) showed that learning the variance schedule improves performance on image density estimation benchmarks. In all previous work, variance scheduling was chosen arbitrarily. In our work instead, the entire spatio-temporal properties of the corruption are determined by the shortest path.

2.3. Image blurring

Most of the previous work on Diffusion Models fixed the spatial corruption to isotropic noise. However, a few recent studies introduced alternative image corruptions, in which different frequencies are degraded at different times (Rissanen et al., 2023; Lee et al., 2022a; Bansal et al., 2022; Daras et al., 2022; Hoogeboom & Salimans, 2023). When higher frequencies are degraded first, the observed effect is image blurring. The work of (Rissanen et al., 2023) proposes simulating the heat equation, given the equivalence of heat dissipation and Gaussian blurring. The work of Hoogeboom & Salimans (2023) combines heat dissipation and additive noise, formalizing it as a diffusion process with anisotropic noise. The work of Lee et al. (2022a) introduced Gaussian blur with monotonically increasing power while following the variance schedule of Ho et al. (2020). The works of Bansal et al. (2022) and Daras et al. (2022) extended Diffusion Models to a wide variety of corruptions, including Gaussian blur. In all these studies, the choice of corruption is arbitrary. The justification for using image blurring is that low-frequency features are more important for human perception of images. In our work instead, we show that the shortest path corresponds to image sharpening and noise deblurring.

2.4. Reverse process

Diffusion models may differ in the parametrization of the reverse process. A neural network may be trained to either predict the noise or the original image. The work of Ho et al. (2020) found that predicting noise results in better performance. However, recent results show that using a combination of the two parametrizations may improve the quality of generated images (Benny & Wolf, 2022). The work of Salimans & Ho (2022) used progressive distillation to skip iterations in the reverse process, and showed that predicting the original image instead of noise achieves the best sampling performance after distillation. We use noise prediction in our work, similar to Ho et al. (2020), but other parameterizations may be implemented as well.

The work of Ma et al. (2022) introduced a matrix preconditioning method to accelarete the reverse process in score-based models. The work of Bao et al. (2022b;a) proposed learning the optimal covariance of the reverse process and showed significant improvements on image quality. Recently, Lu et al. (2022) simplified the expression of the reverse diffusion and significantly improved the speed of generation and quality of images. These features could be added to SPD in the future and are likely to provide further improvements.

2.5. Other improvements

The work of Guth et al. (2022) uses orthogonal wavelets to decompose the images and generates samples in the space of wavelets. In our work, we use a Fourier basis instead of a wavelet basis, and our noising procedure is non-uniform. The work of Lee et al. (2022b) replaces the isotropic Gaussian prior with a distribution of mean and covariance computed on the data. Similar to our work, this approach may shorten the trajectory between the dataset and the prior, but it may not correspond to the shortest path. The work of Khrulkov et al. (2023) shows that diffusion models implement the optimal transport from the data to the target distribution. However, our work is concerned with optimality of the trajectory, rather than of the mapping between the initial and final state.

Other improvements of Diffusion Models that are orthogonal to our work include: The work of Dockhorn et al. (2021), which increased the sampling speed by augmenting the diffusion process with auxiliary variables. The works of Vahdat et al. (2021) and Jing et al. (2022), which propose diffusing in a latent space, improving the generation quality and the computational costs. The work of Watson et al. (2021), which designed a differentiable parametric sampler that can be optimized for fast data generation. All of these can be in principle added to our algorithm and are expected to provide further improvements.

3. Shortest Path Diffusion

In this section, we compute analytically the shortest path for Gaussian distributions, and propose an application to non-Gaussian case, in particular to natural images. We provide the algorithm of Shortest Path Diffusion and discuss its complexity. Details of derivations and proofs are provided in the appendix.

3.1. Shortest path for Gaussian distributions

We consider the simple case of an image $\mathbf{x}$ distributed according to a Gaussian distribution with zero mean and covariance matrix Σ,

$$\mathbf{x} \sim \mathcal{N}\left(\mathbf{0}, \Sigma\right) \tag{1}$$

The vector $\mathbf{x}$ concatenates all pixels of an image, and the matrix Σ includes the covariances of all pairs of pixels. For example, a 32×32 image has 1024 pixels, thus $\mathbf{x}$ is a vector of 1024 elements and Σ is a 1024×1024 matrix. The probability distribution is completely described by Σ, and any transformation from an initial distribution at time

$t = 0$ to a final distribution at time $t = T$ is described by the temporal change in covariance Σ_t.

The sequence Σ_t describes a path in the space of probability distributions. Given the initial Σ_0 and the final Σ_T, what is the shortest path Σ_t? Here we choose to measure path lengths using the Fisher metric, because it provides a bound on the precision of maximum likelihood estimation of probability distributions via the Cramer-Rao theorem (Amari, 2016). The reason is that Diffusion Models are trained by the Evidence Lower Bound (ELBO), which is a bound on the likelihood (Sohl-Dickstein et al., 2015; Ho et al., 2020). We also highlight that the Fisher metric is invariant for reparameterization of probability distributions (Amari, 2016), therefore any measured path length (and thus the shortest path) does not depend on the chosen parameterization of the distribution (for example, using the precision instead of the covariance, or any other invertible transformation of the covariance).

Theorem 3.1. *Given two Gaussian distributions with zero mean and covariance matrix equal to, respectively, Σ_0 and Σ_1, where Σ_1 is non-singular. Given the Riemannian metric defined by the Fisher information, the shortest path between the two distributions is given by*

$$\Sigma_t = \Sigma_1^{1/2} \left(\Sigma_1^{-1/2} \Sigma_0 \Sigma_1^{-1/2} \right)^{1-t} \Sigma_1^{1/2} \tag{2}$$

where $t \in (0, 1)$ measures the relative distance travelled along the path.

Proof is provided in appendix A (see also Pinele et al. (2020)). For numerical purposes, we implement a discrete-time version of the shortest path, with $t = 0, 1, 2, ..., T$. Consistent with previous work, we set the final distribution as isotropic Gaussian, thus the final covariance is equal to $\Sigma_T = \mathrm{I}$ (identity matrix) for any initial covariance Σ_0. In the shortest path, each eigenvalue σ^2 of the covariance matrix, representing the variance of a given combination of pixels, evolves according to

$$\sigma_t^2 = \left(\sigma_0^2 \right)^{1-t/T} \tag{3}$$

where σ_0^2 is the initial variance. Figure 1 (top) shows an example with two variances, where the large one drops first and the smaller one raises later. We stress that the metric is not Euclidean therefore the linear path in figure 1 is not the shortest path. The exponential dependence on time in equation 3 implies that the rate of change depends on the initial variance: small σ_0^2 change slowly, while large σ_0^2 change faster.

In matrix form, the shortest path corresponds to the following covariance schedule

$$\Sigma_t = F D^{1-t/T} F^{\dagger} \tag{4}$$

where F is the matrix of orthogonal eigenvectors of Σ_0 and D is the diagonal matrix of positive eigenvalues of Σ_0. Here $\dagger$ denotes the operations of matrix transpose and complex conjugation, and $D^{1-t/T}$ is the diagonal matrix where each element is raised to the power of $1 - t/T$.

Equation 4 describes the shortest path between a Gaussian distribution with covariance Σ_0 and an isotropic Gaussian. In the context of diffusion models, this is a corruption procedure that starts from the data distribution, which has a rich structure described by the covariance Σ_0, and terminates with pure noise (isotropic Gaussian). This corruption procedure is implemented by a forward process that corrupts individual images sampled from the data distribution (Ho et al., 2020). In the next section, we derive a corruption procedure implementing the shortest path.

3.2. Image corruption

The following theorem provides a data corruption procedure implementing the shortest path of equation 4.

Theorem 3.2. *Given a random vector $\mathbf{x}_0$ of zero mean and covariance Σ_0, and another random vector $\boldsymbol{\epsilon}_t$ of zero mean and covariance I (isotropic), where $\mathbf{x}_0$ and $\boldsymbol{\epsilon}_t$ are uncorrelated. Assume the matrix $(I - \Sigma_0)$ is invertible. Define the matrix $\Phi_t = (I - \Sigma_0^{1-t})(I - \Sigma_0)^{-1}$ and note that, for $t \in (0, 1)$, Φ_t and $(\mathrm{I} - \Phi_t)$ are positive definite and, respectively, monotonically decreasing and increasing functions of Σ_0. Then, the corrupted vector $\mathbf{x}_t$ defined by*

$$\mathbf{x}_t = \Phi_t^{\frac{1}{2}} \mathbf{x}_0 + (I - \Phi_t)^{\frac{1}{2}} \boldsymbol{\epsilon}_t \tag{5}$$

has zero mean and covariance equal to Σ_0^{1-t}.

Proof is provided in appendix B. Therefore, shortest path is implemented by corrupting images from $\mathbf{x}_0$ to $\mathbf{x}_T$ according to equation 5, where $\boldsymbol{\epsilon}_t$ is sampled from an isotropic Gaussian. Φ_t is a matrix, thus the image $\mathbf{x}_0$ is corrupted by a linear transformation (instead of a simple rescaling (Ho et al., 2020)). The noise $\boldsymbol{\epsilon}_t$ is also linearly transformed. The linear transform Φ_t is equal to

$$\Phi_t = (\mathrm{I} - \Sigma_0^{1-t/T})(\mathrm{I} - \Sigma_0)^{-1}. \tag{6}$$

Equation 5 is similar to the forward process of a few recent studies (Rissanen et al., 2023; Lee et al., 2022a; Bansal et al., 2022; Daras et al., 2022; Hoogeboom & Salimans, 2023). However, those studies picked an arbitrary form of the matrix Φ_t and tried to optimize it empirically. Instead, our work provides an optimal form, given by equation 6.

3.3. Application to real images

The distribution of real images is not Gaussian, therefore the shortest path of section 3.1 does not apply. However, its covariance matrix Σ_0 has a rich structure describing the second-order statistics of all pairs of pixels. We propose to approximate the shortest path between the distribution of real images and an isotropic Gaussian by corrupting images with the forward process of equations 5, 6. We note that, even if the distribution of real images is not Gaussian, the forward process 5, 6 still implies the covariance schedule 4. For non-Gaussian distributions, it is unknown whether the shortest path has covariance schedule 4, but we hypothesize that this forward process provides a good approximation to the true shortest path.

The application of equations 5, 6 to real images requires computing the covariance matrix Σ_0 of their distribution. However, this computation may be expensive, for example 1024×1024 images have a 1049600×1049600 covariance matrix. Fortunately, the form of the covariance matrix of translation invariant distributions is known to be equal to

$$\Sigma_0 = FDF^\dagger \tag{7}$$

where F is the 2-dimensional Discrete Fourier Transform (DFT) matrix, and D is a diagonal matrix with the power spectrum of the data. We work under the assumption that natural images are approximately translation invariant (this may not apply to certain datasets, e.g. centered faces, CelebA). Therefore, the eigenvectors of Σ_0 are given by the DFT matrix, and its eigenvalues are given by the power spectrum.

The power spectrum of natural images is also known to decrease with the squared frequency norm (Hyvärinen et al., 2009). Thus, we model the power spectrum with the following equation

$$D_{ii} = \frac{c_1}{|c_2 + f_i|^m} \tag{8}$$

where f_i is the frequency corresponding to index i, and is equal to the norm of the vector of frequencies along the horizontal and vertical axes of an image

$$f = \sqrt{f_x^2 + f_y^2} \tag{9}$$

We set the exponent m equal to 2 in most of our experiments, following (Hyvärinen et al., 2009), while we fit the constants c_1, c_2 on the empirical power spectrum of the dataset. Figure 2 shows the power spectrum computed for CIFAR10 (Krizhevsky, 2009) and ImageNet 64×64 (Deng et al., 2009) datasets. We show in section 4.5 (see figure 4) that using any values of m different from $m = 2$ results in worse performance, suggesting that the shortest path is the optimal procedure for corrupting images.

Algorithm 1 Shortest Path Diffusion (batch size = 1)

```
Given: dataset and randomly initialized network $g_\theta$
Compute power spectrum of dataset
Fit $c_1, c_2$ on power spectrum with model (8)
Compute optimal filter $\Psi_t$ for all $t = 1 : T$ (13)
while not converged do
    Sample $\mathbf{x}_0$ from dataset and compute its DFT $\mathbf{u}_0$
    Sample $t$ uniformly in $1 : T$
    Sample noise $\boldsymbol{\epsilon}_t$ and compute its DFT $\boldsymbol{\xi}_t$
    Compute corrupted $\mathbf{u}_t$ (12) and its inverse DFT $\mathbf{x}_t$
    One-step optimization of $\theta$ with loss$(g_\theta(\mathbf{x}_t), \boldsymbol{\epsilon}_t)$
end while
```

3.4. Algorithm and complexity

Training of Shortest Path Diffusion is described in algorithm 1. We note that, in general, equation 5 requires a large amount of memory and compute, due to the quadratic scaling of Φ_t with the dimension of data d (number of pixels). However, in this section we provide an implementation that scales linearly in the case of real images, which neither require computation of F nor any $d \times d$ matrix multiplication.

Similar to the work of Lee et al. (2022a) and Hoogeboom & Salimans (2023), we corrupt images in frequency space instead of pixel space. We denote by $\mathbf{u}_t$ the 2-dimensional DFT of image $\mathbf{x}_t$, equal to

$$\mathbf{u}_t = F^\dagger \mathbf{x}_t \tag{10}$$

Given the transformed $\mathbf{u}_t$, we can recover the image by just using inverse Fourier transform

$$\mathbf{x}_t = F\mathbf{u}_t \tag{11}$$

Note that the complexity of DFT is quasilinear in d (log-linear). Application of equations 5, 6 in frequency space is given by

$$\mathbf{u}_t = \Psi_t^{\frac{1}{2}} \mathbf{u}_0 + (\mathrm{I} - \Psi_t)^{\frac{1}{2}} \boldsymbol{\xi}_t \tag{12}$$

$$\Psi_t = (\mathrm{I} - D^{1-t/T})(\mathrm{I} - D)^{-1} \tag{13}$$

where $\boldsymbol{\xi}_t$ is noise in frequency space, $\boldsymbol{\xi}_t = F^\dagger \boldsymbol{\epsilon}_t$. The matrix Ψ in equation 13 is diagonal, thus equation 12 can be implemented by element-wise multiplication and does not require any $d \times d$ matrix multiplication. Fitting the power spectrum of the data also scales linearly (see appendix C), thus overall complexity of the algorithm is linear in d.

Algorithm 1 implements a batch size equal to one, but it is straightforward to implement it for larger batch sizes. We use the *simple* loss function defined in Ho et al. (2020), which trains a neural network g_θ to estimate the mapping $\hat{\boldsymbol{\epsilon}}_t = g_\theta(\mathbf{x_t})$.

Algorithm 2 Image generation (reverse process)

```
**Given:** trained neural network $g_\theta$ and optimal filter $\Psi_t$
Set noise $\sigma_t$ for all $t = 1 : T$
Set $t = T$
Sample $\mathbf{x}_T \sim \mathcal{N}(0, \mathrm{I})$ and compute its DFT $\mathbf{u}_T$
**while** $t > 0$ **do**
    Sample $\mathbf{z}_t \sim \mathcal{N}(0, \mathrm{I})$
    Compute $\mathbf{u}_{t-1}$ (14) and its inverse DFT $\mathbf{x}_{t-1}$
    $t = t - 1$
**end while**
**Return** $\mathbf{x}_0$
```

3.5. Image generation

Algorithm 2 describes the algorithm for image generation (reverse process). For generation of images we essentially follow Ho et al. (2020). However, similar to the forward process, the reverse process also runs in frequency space, as in recent works (Lee et al., 2022a; Hoogeboom & Salimans, 2023). After training a neural network on the mapping $\hat{\boldsymbol{\epsilon}}_t = g_\theta(\mathbf{x_t})$ by Shortest Path Diffusion, we use it to approximate the reverse process according to

$$\begin{aligned}\mathbf{u}_{t-1} &= \Psi_t^{-\frac{1}{2}} \Psi_{t-1}^{\frac{1}{2}} \mathbf{u}_t + \sigma_t F^\dagger \mathbf{z}_t \\ &- \Psi_t^{-\frac{1}{2}} \Psi_{t-1}^{\frac{1}{2}} (\mathrm{I} - \Psi_t \Psi_{t-1}^{-1})(\mathrm{I} - \Psi_t)^{-\frac{1}{2}} F^\dagger g_\theta(\mathbf{x}_t)\end{aligned} \tag{14}$$

where Ψ_t is diagonal and applies element-wise, $\mathbf{z}_t$ is isotropic Gaussian noise and σ_t is chosen depending on T (also diagonal, see section 4).

While the neural network operates in pixel space, we use DFT to compute the transformed estimate and run the reverse process in frequency space. This allows using previously successful neural network architectures, which are known to operate well in pixel space.

4. Experiments

In this section, we validate empirically our proposed Shortest Path Diffusion (SPD) on unconditional image generation.

We use algorithm 1 for training and algorithm 2 for generating images, as described in section 3. We conduct a range of experiments and show that the shortest path leads to the best quality of generated images in comparison to similar methods. Our code is available at `https://github.com/mtkresearch/shortest-path-diffusion`

4.1. Dataset and metrics

Here we describe the dataset and metrics used for our experimentation. We use CIFAR10 (Krizhevsky, 2009) and ImageNet (Deng et al., 2009), two of the most frequently used benchmarks for evaluating generative models on images. CIFAR10 has resolution of 32×32 pixels (dimension $d = 1024$), while for ImageNet we use images scaled to 64×64 resolution (dimension $d = 4096$). For both datasets, we only consider the task of *unconditional* image generation.

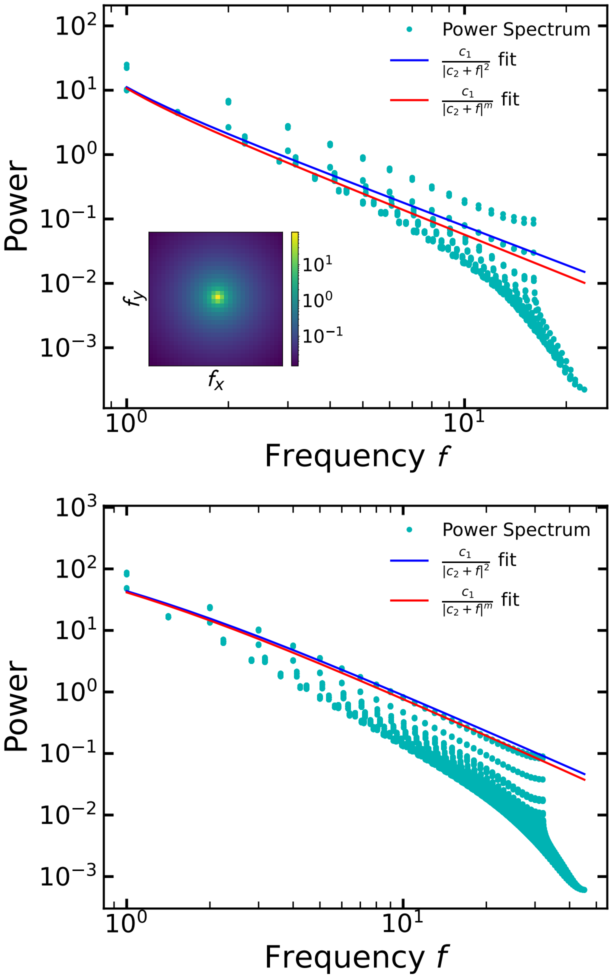

Figure 2. **Power spectrum and model fit**. Dots show the empirical power spectrum, lines are fits provided by the model in equation 8. **Top**: CIFAR10 power spectrum, with $c_1 = 7.7$ and $c_2 = -0.3$. Inset: The 2-dimensional power spectrum obtained from the fit, as a function of the horizontal (f_x) and vertical (f_y) frequency of images. **Bottom**: ImageNet 64x64 power spectrum, with $c_1 = 96.79$ and $c_2 = 0.49$. We also fit a model $\sim 1/f^m$, finding $m = 2.1$ for CIFAR10 and $m = 2.05$ for ImageNet (while fixing $c_2 = 0.49$), suggesting that the inverse square model is accurate.

Individual pixels are re-scaled to the range of $[-1, 1]$ following the usual practice in the literature (Ho et al., 2020; Dhariwal & Nichol, 2021). We evaluate the quality of generated images by Frécet Inception Distance (FID) (Heusel et al., 2017). We use standard practice for evaluating FID, comparing generated samples with real data and using the same

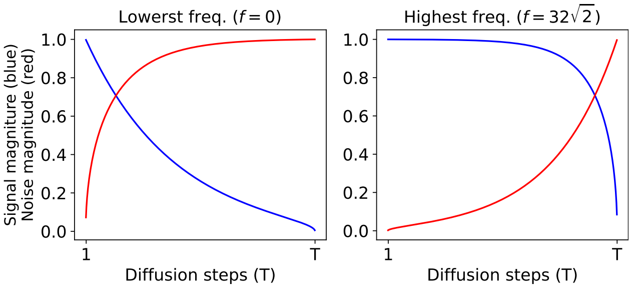

Figure 3. **Temporal dynamics of corruption for different frequencies**. Low frequencies change earlier (**left**), while high frequencies change later (**right**) during the corruption procedure of CIFAR10 images, for both the image (signal, blue) and the noise (red). Figure 1 (bottom) shows the corruption of an example image.

Inception checkpoints as in Nichol & Dhariwal (2021a); Dhariwal & Nichol (2021). We use $50,000$ samples for CIFAR10 and $10,000$ samples for ImageNet 64×64, following Nichol & Dhariwal (2021a).

4.2. Power spectrum

The optimal corruption filter of SPD is obtained by the power spectrum of the dataset. We compute the power spectrum of each image in the training set and we average the power spectrum across all images, separately for each channel. Figure 2 shows the power spectrum against frequency for CIFAR10 (top) and ImageNet 64×64 (bottom). For each value of the horizontal axis we obtain multiple values of the power spectrum, corresponding to different channels and different directions of the frequency vector. Note that frequencies are 2-dimensional vectors, the horizontal axes of figure 2 show their norm.

We fit parameters c_1 and c_2 of the model in equation 8 using least squares regression. For CIFAR10, We obtain $c_1 = 7.7$ and $c_2 = -0.3$, while for ImageNet 64×64 we obtain $c_1 = 96.79$ and $c_2 = 0.49$. These values are used to compute the optimal corruption filter by equation 13. Also shown in figure 2, we fit the exponent m in equation 8, obtaining a value of 2.1 for CIFAR10 and 2.05 for ImageNet 64×64. This confirms that the inverse square law, i.e. $m = 2$, is a good model of the spectrum of natural images (Hyvärinen et al., 2009).

4.3. Optimal corruption filter

In this section, we investigate how images are affected by the optimal corruption filter obtained in section 4.2. The linear filter Ψ_t progressively dissipates the original image through equation 12, but different frequencies of the original image dissipate at different times. Similarly, different frequencies of the noise perturb the image at different times.

Figure 3 shows the temporal change of signal and noise at different frequencies during corruption of CIFAR10 images. We observe that lower frequencies dissipate first and higher frequencies dissipate later. Simultaneously, lower frequencies of the noise appear first, while higher frequencies appear later. This corresponds to image sharpening and noise deblurring, and is a general property of the shortest path because equation 8 is a decreasing function of f and equation 13 is a decreasing function of D (for $t \in [1, N]$). Figure 1 (bottom) shows the corruption of an example image.

We highlight that Shortest Path Diffusion completely determines the change of signal and noise during image corruption. All previous studies have arbitrarily set a variety of schedules for signal and noise and tried to hyper-optimize them. In our work, the signal and noise schedules are fixed by the optimal spatio-temporal filter Ψ_t.

4.4. Training and sampling

We use a slight modification of the codebase in Dhariwal & Nichol (2021). The only difference is the corruption procedure, that we implement according to our SPD algorithm, equipped with the optimal corruption filter obtained in section 4.2. For the neural network g_θ, we use the same variant of UNet as in (Dhariwal & Nichol, 2021), without any modification. We optimize parameters θ by minimizing the *simple* loss (Ho et al., 2020). We used Adam optimizer with learning rate of 1×10^{-4}.

For CIFAR10, we use batch size 1024, $150,000$ training iterations, and we record model checkpoints every $5,000$ iterations. For ImageNet 64×64, we use batch size 336, 1M training iterations, and we record model checkpoints every $3,000$ iterations. We report the best FID score across checkpoints. Similar to Ho et al. (2020), for generating images we use Eq. 14 with $\sigma_t = (\mathrm{I} - \Psi_t \Psi_{t-1}^{-1})$ for $T > 300$ and $\sigma_t = (\mathrm{I} - \Psi_{t-1})(\mathrm{I} - \Psi_t)^{-1}(\mathrm{I} - \Psi_t \Psi_{t-1}^{-1})$ for $T \leq 300$, where T is the number of diffusion steps.

We compare SPD with other methods running on the same codebase: iDDPM and iDDPM+DDIM (Nichol & Dhariwal, 2021b; Song et al., 2021). Our implementation of iDDPM runs on the codebase of Dhariwal & Nichol (2021) and gives slightly better results than the original (Nichol & Dhariwal, 2021b). We re-train SPD and iDDPM for each different value of T, while iDDPM+DDIM uses the deterministic DDIM sampler and is trained once at $T = 4,000$ (Song et al., 2021).

4.5. Results

First, we test our main hypothesis that Shortest Path Diffusion provides the optimal corruption. As discussed in section 3.3, the optimal SPD filter depends on the power

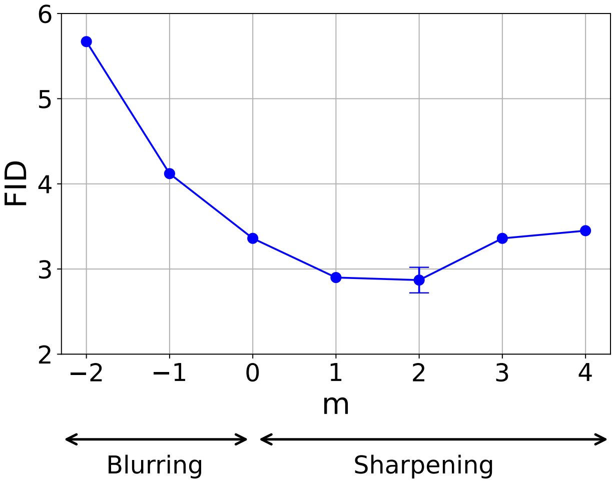

Figure 4. **Any deviation from Shortest Path Diffusion deteriorates quality of CIFAR10 images**. Image quality is measured by FID (lower is better). SPD corresponds to the value $m = 2$ as shown in the power spectrum of figure 2. For comparison, we also run other corruptions, corresponding to other m values. Negative and positive values of exponent m result, respectively, in image blurring and image sharpening, while $m = 0$ corresponds to uniform noising of all the frequencies. We found that image quality is worse in all other cases, suggesting that SPD provides the optimal corruption. We used $T = 500$ diffusion timesteps in all experiments. We run 5 experiments with different initialization for $m = 2$, where standard deviation is shown, while we have single runs for all other values of m.

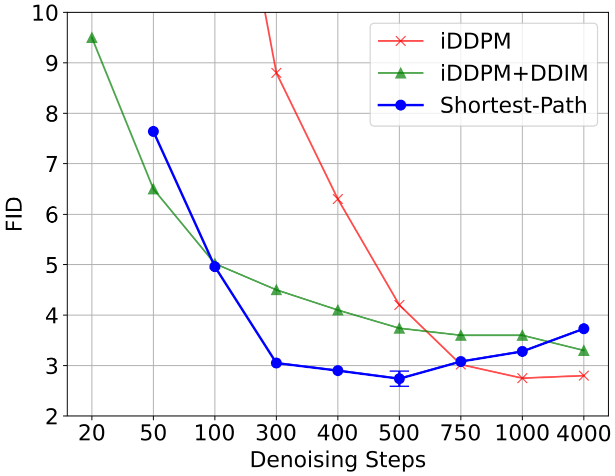

Figure 5. **CIFAR10 image quality against total diffusion timesteps**. Image quality is measured by FID (lower is better). All algorithms (iDDPM, iDDPM+DDIM, SPD) run in the same codebase. The only difference between SPD and iDDPM is the corruption procedure. SPD outperforms iDDPM+DDIM in the interval of $100 \leq T \leq 1000$ and iDDPM for $T \leq 500$.

spectrum of the data, in particular the exponent m in the model of equation 8, which is equal to $m = 2$ for natural images. According to our hypothesis, any other value of this exponent should result in worse performance, because it would determine a different filter and therefore a different corruption procedure. To test the optimality of shortest path, we changed m in the range $[-2, 4]$ for models trained on CIFAR10. Negative and positive values of the exponent m result, respectively, in image blurring and image sharpening, while $m = 0$ corresponds to uniform noising of all the frequencies. Figure 4 shows that the best performance is obtained for $m = 2$, as predicted by our hypothesis. We only test a subset of possible values for m, but results suggest that $m = 2$ is nearly optimal. To obtain filters at different values of m, we fixed c_2 and we set c_1 for each value of m such that the noise variance for all filters at half-time of the forward process is equal.

We also compare our best SPD model with other similar approaches, specifically, methods with forward noising processes containing blurring or a mixture of blurring and noising. We show in table 1 that SPD outperforms all of other methods. Again, these other approaches provide only a small subset of all possible corruption filters, but our hypothesis that the optimal SPD filter provides the best corruption procedure still stands. We stress that, although all methods in table 1 corrupt frequencies at different speeds, SPD sharpens images instead of blurring them.

In figure 5, we evaluate SPD across different values of the number of total diffusion timesteps T. This number is particularly important since a smaller T allows generating images faster. We stress that, in this implementation, the only difference between SPD and iDDPM is the corruption procedure, everything else is equal (codebase, hyperparameters, computing machines). SPD provides good image quality in a wide range of values of T, suggesting that it is resilient against changes of T values, and outperforms iDDPM when $T \leq 500$. For very large T, we do not expect SPD to give any advantage because the errors of the reverse process may

Table 1. **Comparison of SPD with methods based on image blurring, for CIFAR10 dataset**. Image quality is measured by FID (lower is better). SPD uses $T = 500$ diffusion timesteps, for all other methods we show the best FID reported by the authors in Daras et al. (2022) (Soft Diffusion) and Hoogeboom & Salimans (2023) (Blurring Diffusion).

Methods	FID
Soft Diffusion	4.64
Blurring Diffusion	3.17
SPD (Ours)	**2.74**

in principle vanish. These expectations are confirmed by observing that the advantage of SPD with respect to iDDPM occurs especially at lower values of T.

We also compare SPD against iDDPM+DDIM (Song et al., 2021), because the latter is expected to provide better results than iDDPM at smaller T values. SPD outperforms iDDPM+DDIM for a range of values $100 \leq T \leq 1000$. Although SPD does not outperform iDDPM+DDIM below $T < 100$, the FID for both methods are poor enough to be discarded. Figure 7 in appendix D shows example images generated by SPD, iDDPM and iDDPM+DDIM.

Lastly, in table 2 we show quality of *unconditional* generation of ImageNet 64×64 images, in comparison with iDDPM (Nichol & Dhariwal, 2021a). Our model outperforms iDDPM despite having a lower number of diffusion steps and training for fewer iterations. We stress that FIDs in table 2 correspond to *unconditional* generation, which are higher (worse) than FIDs for *conditional* generation reported in other studies. Figure 8 in appendix D shows example images generated from the model.

5. Conclusions

We introduced Shortest Path Diffusion (SPD), a Diffusion Model providing a unique procedure for data corruption. Previous work explored different procedures for corrupting data and tried to optimize them empirically. Instead, we argue that SPD provides the optimal corruption since it minimizes the length of the path taken by the corruption in the space of probability distributions. Although we do not provide any proof of optimality, we argue that taking the shortest path may reduce the effect of errors in estimating the reverse transition probabilities. Interestingly, while previous work explored image blurring, instead we found that image sharpening provides better results.

In contrast to previous work, the corruption of SPD is data-dependent, thus SPD provides the flexibility of adjusting the corruption to the given dataset (but see Lee et al. (2022b) for a similar approach). Furthermore, SPD can be applied not only to images but also to other types of data. However, different types of data may require a slightly different treatment in order to make SPD feasible. In general, SPD requires computing the full covariance matrix of the data, which scales quadratically with its dimension. However, this complexity can be reduced to linear if a strong prior on the form of the covariance is given.

For natural images, whose distribution is approximately translation invariant, SPD requires computing the power spectrum of the data, which scales linearly with either the dimension or the size of dataset. The same approach may apply to several other image datasets (e.g. LSUN). Furthermore, a similar approach may apply to non-image data that is nevertheless translation-invariant, such as audio and speech data. However, other types of data may require a different treatment, including non-translation invariant image datasets (e.g. CelebA).

A limitation of our work is that the shortest path is computed in closed form for Gaussian distributions only, while most distributions of interest, including real images, are not Gaussian. We hypothesized that the shortest path for real images could be approximated by that of a Gaussian distribution with the same covariance. We also chose the Fisher metric to compute the shortest path, but other choices are possible (e.g. Wasserstein metric, see Khrulkov et al. (2023)). We found strong empirical support for the assumptions of this study, but more research is required to test them further, for example on higher-resolution images, other types of data or metrics.

SPD mostly concerns the corruption procedure, it is orthogonal to studies that improved Diffusion Models by other means, and those studies may be used to further improve SPD. Those include, for example, using dedicated ODE solvers for sampling from the model (Lu et al., 2022), more accurate estimation of the covariance of the reverse transition probability (Bao et al., 2022b;a), using auxiliary variables (Dockhorn et al., 2021), learning the optimal reverse steps (Salimans & Ho, 2022), and diffusing in latent space (Vahdat et al., 2021; Jing et al., 2022). Some of this studies obtain FID scores lower (better) than our work, but we believe that they could be improved further by implementing our proposed SPD corruption. We believe that SPD provides a useful tool to advance the progress of Diffusion Models in generating a variety of different types of data.

Table 2. ***Unconditional*** **generation of ImageNet 64x64 images**. FID evaluations are based on 10,000 generated samples (lower is better). Results for iDDPM are copied from Nichol & Dhariwal (2021a). We stress that these numbers correspond to *unconditional* generation, which are higher (worse) than FIDs for *conditional* generation reported in other studies.

Methods	Diffusion steps	Training steps	FID
iDDPM	4000	1.5M	19.2
SPD (Ours)	1000	1M	**13.7**

References

Amari, S.-i. *Information geometry and its applications*, volume 194. Springer, 2016.

Bansal, A., Borgnia, E., Chu, H.-M., Li, J. S., Kazemi, H., Huang, F., Goldblum, M., Geiping, J., and Goldstein, T. Cold diffusion: Inverting arbitrary image transforms without noise, 2022.

Bao, F., Li, C., Sun, J., Zhu, J., and Zhang, B. Estimating the optimal covariance with imperfect mean in diffusion probabilistic models. *arXiv preprint arXiv:2206.07309*, 2022a.

Bao, F., Li, C., Zhu, J., and Zhang, B. Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. *arXiv preprint arXiv:2201.06503*, 2022b.

Bengio, Y., Laufer, E., Alain, G., and Yosinski, J. Deep generative stochastic networks trainable by backprop. In *International Conference on Machine Learning*, pp. 226–234. PMLR, 2014.

Benny, Y. and Wolf, L. Dynamic dual-output diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11482–11491, 2022.

Bond-Taylor, S., Leach, A., Long, Y., and Willcocks, C. G. Deep generative modelling: A comparative review of vaes, gans, normalizing flows, energy-based and autoregressive models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7327–7347, 2022. doi: 10.1109/TPAMI.2021.3116668.

Chen, R. T., Behrmann, J., Duvenaud, D. K., and Jacobsen, J.-H. Residual flows for invertible generative modeling. *Advances in Neural Information Processing Systems*, 32, 2019.

Child, R. Very deep vaes generalize autoregressive models and can outperform them on images. *ICLR*, 2021.

Child, R., Gray, S., Radford, A., and Sutskever, I. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

Daras, G., Delbracio, M., Talebi, H., Dimakis, A. G., and Milanfar, P. Soft diffusion: Score matching for general corruptions. *arXiv preprint arXiv:2209.05442*, 2022.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. Density estimation using real nvp. *ICLR*, 2017.

Dockhorn, T., Vahdat, A., and Kreis, K. Score-based generative modeling with critically-damped langevin diffusion. *arXiv preprint arXiv:2112.07068*, 2021.

Fox, C. *An introduction to the calculus of variations*. Courier Corporation, 1987.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

Guth, F., Coste, S., Bortoli, V. D., and Mallat, S. Wavelet score-based generative modeling. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=xZmjH3Pm2BK.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 2017.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Hoogeboom, E. and Salimans, T. Blurring diffusion models. *ICLR*, 2023.

Horn, R. A. and Johnson, C. R. *Matrix Analysis*. Cambridge University Press, Cambridge; New York, 2nd edition, 2013. ISBN 9780521839402.

Hyvärinen, A., Hurri, J., and Hoyer, P. O. *Natural image statistics: A probabilistic approach to early computational vision.*, volume 39. Springer Science & Business Media, 2009.

Jing, B., Corso, G., Berlinghieri, R., and Jaakkola, T. Subspace diffusion generative models. *arXiv preprint arXiv:2205.01490*, 2022.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, 2020.

Khrulkov, V., Ryzhakov, G., Chertkov, A., and Oseledets, I. Understanding ddpm latent codes through optimal transport. *ICLR*, 2023.

Kingma, D. and Welling, M. Auto-encoding variational bayes. *ICLR*, 2014.

Kingma, D., Salimans, T., Poole, B., and Ho, J. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021.

Krizhevsky, A. Learning multiple layers of features from tiny images. *Master's thesis, University of Tront*, 2009.

Lee, S., Chung, H., Kim, J., and Ye, J. C. Progressive deblurring of diffusion models for coarse-to-fine image synthesis. *arXiv preprint arXiv:2207.11192*, 2022a.

Lee, S.-G., Kim, H., Shin, C., Tan, X., Liu, C., Meng, Q., Qin, T., Chen, W., Yoon, S., and Liu, T.-Y. Priorgrad: Improving conditional denoising diffusion models with data-dependent adaptive prior. In *International Conference on Learning Representations*, 2022b. URL https://openreview.net/forum?id=_BNiN4IjC5.

Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., and Zhu, J. DPM-solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=2uAaGwlP_V.

Ma, H., Zhang, L., Zhu, X., and Feng, J. Accelerating score-based generative models with preconditioned diffusion sampling. In *European Conference on Computer Vision*, pp. 1–16. Springer, 2022.

Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., and Chen, M. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *International Conference on Machine Learning*, 2022.

Nichol, A. Q. and Dhariwal, P. Improved denoising diffusion probabilistic models. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8162–8171, 2021a.

Nichol, A. Q. and Dhariwal, P. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pp. 8162–8171. PMLR, 2021b.

Pinele, J., Strapasson, J. E., and Costa, S. I. The fisher–rao distance between multivariate normal distributions: Special cases, bounds and applications. *Entropy*, 22(4): 404, 2020.

Rissanen, S., Heinonen, M., and Solin, A. Generative modelling with inverse heat dissipation. *ICLR*, 2023.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E., Ghasemipour, S. K. S., Ayan, B. K., Mahdavi, S. S., Lopes, R. G., et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.

Salimans, T. and Ho, J. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

Salimans, T., Kingma, D., and Welling, M. Markov chain monte carlo and variational inference: Bridging the gap. In *International conference on machine learning*, pp. 1218–1226. PMLR, 2015.

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

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. *ICLR*, 2021.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

Song, Y. and Ermon, S. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Vahdat, A., Kreis, K., and Kautz, J. Score-based generative modeling in latent space. *Advances in Neural Information Processing Systems*, 34:11287–11302, 2021.

Van Den Oord, A., Kalchbrenner, N., and Kavukcuoglu, K. Pixel recurrent neural networks. In *International conference on machine learning*, pp. 1747–1756. PMLR, 2016.

Watson, D., Chan, W., Ho, J., and Norouzi, M. Learning fast samplers for diffusion models by differentiating through sample quality. In *International Conference on Learning Representations*, 2021.

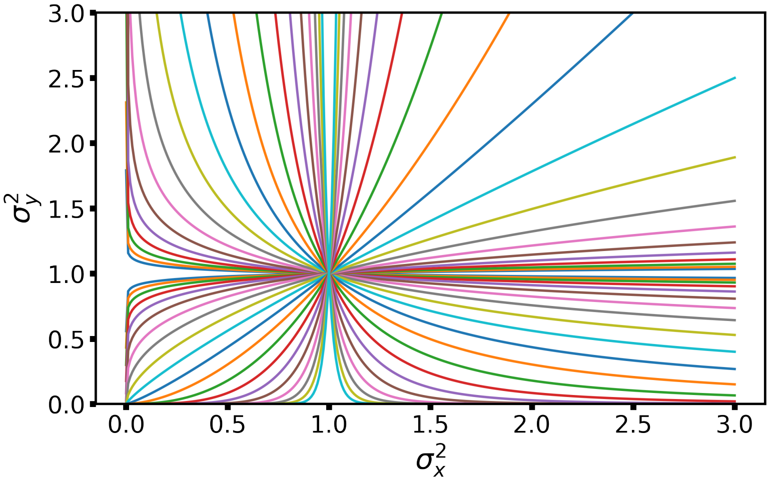

Figure 6. **Shortest path between Gaussians using Fisher metric.** Eigenvalues σ_x^2 and σ_y^2 of the covariance matrix follow equation 17.

A. Shortest path for Gaussian distributions

Theorem A.1. *Given two Gaussian distributions with zero mean and covariance matrix equal to, respectively, Σ_0 and Σ_1, where Σ_1 is non-singular. Given the Riemannian metric defined by the Fisher information, the shortest path between the two distributions is given by*

$$\Sigma_t = \Sigma_1^{1/2} \left(\Sigma_1^{-1/2} \Sigma_0 \Sigma_1^{-1/2} \right)^{1-t} \Sigma_1^{1/2} \tag{15}$$

where $t \in (0, 1)$ measures the relative distance travelled along the path.

Remark A.2. In the special case of $\Sigma_1 = I$, theorem A.1 implies that

$$\Sigma_t = \Sigma_0^{1-t} \tag{16}$$

Therefore the eigenvectors of Σ_t are equal to those of Σ_0, and the eigenvalues of Σ_t are equal to those of Σ_0 raised to the power of $1 - t$. Denoting by σ_x^2 and σ_y^2 any pair of eigenvalues of Σ_t, along the shortest path they satisfy the following equation

$$\sigma_y^2 = \left(\sigma_x^2\right)^{\alpha} \tag{17}$$

where α depends on the eigenvalues at $t = 0$. Shortest paths are illustrated in figure 6.

Proof. Theorem A.1 is discussed in Pinele et al. (2020) and references therein, here we provide an alternative derivation for completeness. The length of a curve is described by the sum of the lengths of its infinitesimal segments, where the length of a given segment $d\mathbf{x}$ is equal to its Eucledian norm $|d\mathbf{x}|$. The length of a curve $\mathbf{x}(s) \in \mathbb{R}^d$, parameterized by a scalar s, is equal to the line integral

$$\mathcal{D} = \int_{s_0}^{s_1} ds \left| \frac{d\mathbf{x}}{ds} \right| = \int_{s_0}^{s_1} ds \left(\sum_{i=1}^{d} \frac{dx_i}{ds}^2 \right)^{1/2} \tag{18}$$

where the initial and final points of the curve are represented by $\mathbf{x}(s_0)$ and $\mathbf{x}(s_1)$, respectively. When measuring distances using a non-Euclidean (Riemannian) metric tensor $\mathcal{I} \in \mathbb{R}^{d\times d}$, the length of the curve is equal to

$$\mathcal{D} = \int_{s_0}^{s_1} ds \left(\sum_{i,j=1}^{d} \frac{dx_i}{ds} \mathcal{I}_{ij} \frac{dx_j}{ds} \right)^{1/2} \tag{19}$$

In the case of Gaussian distributions with zero mean, the curve is represented by the covariance $\Sigma(s) \in \mathbb{R}^{d\times d}$. The

Riemannian metric is given by the tensor $\mathcal{I} \in \mathbb{R}^{(d\times d)\times(d\times d)}$ and the length of the curve is given by

$$\mathcal{D} = \int_{s_0}^{s_1} ds \, \left(\sum_{i,j,k,l}^{1,d} \frac{d\Sigma_{ij}}{ds} \mathcal{I}_{ij,kl} \frac{d\Sigma_{kl}}{ds} \right)^{1/2} \tag{20}$$

We use the Fisher information matrix as the metric tensor. For a Gaussians distribution with zero mean and covariance Σ, the Fisher information matrix is equal to

$$F_{ij,kl} = \frac{1}{2} \left(\Sigma^{-1}\right)_{il} \left(\Sigma^{-1}\right)_{jk} \tag{21}$$

Thus, the length of the curve is equal to

$$\mathcal{D} = \frac{1}{\sqrt{2}} \int_{s_0}^{s_1} ds \, \text{Tr} \left(\Sigma^{-1} \frac{d\Sigma}{ds} \Sigma^{-1} \frac{d\Sigma}{ds} \right)^{1/2} \tag{22}$$

It is important to note that, given the properties of the Fisher metric, the length of the curve is *reparameterization invariant*. This means that it depends only on the distribution and not on how the distribution is parameterized (Amari, 2016).

Our aim is to find the curve of shortest length, namely the curve $\Sigma(s)$ that minimizes equation 22. All stationary points of 22 satisfy the Euler-Lagrange equation, which is given by (Fox, 1987)

$$\frac{\partial \mathcal{L}}{\partial \Sigma} = \frac{d}{ds} \frac{\partial \mathcal{L}}{\partial \dot{\Sigma}} \tag{23}$$

where $\dot{\Sigma} = \frac{d\Sigma}{ds}$ and the Lagrangian is equal to

$$\mathcal{L}(\Sigma(s), \dot{\Sigma}(s)) = \frac{1}{\sqrt{2}} \text{Tr}(\Sigma^{-1}\dot{\Sigma}\, \Sigma^{-1}\dot{\Sigma})^{1/2} \tag{24}$$

After taking gradients of the Lagrangian with respect to Σ and $\dot{\Sigma}$, equation 23 becomes

$$-\frac{1}{2\mathcal{L}} \Sigma^{-1}\dot{\Sigma}\Sigma^{-1}\dot{\Sigma}\Sigma^{-1} = \frac{d}{ds} \left(\frac{1}{2\mathcal{L}} \Sigma^{-1}\dot{\Sigma}\Sigma^{-1} \right) \tag{25}$$

This equation can be simplified by a sequence of steps. First, we multiply both sides by the scalar $\frac{2}{\mathcal{L}}$, and we obtain

$$-\Sigma^{-1} \frac{d\Sigma}{\mathcal{L}ds} \Sigma^{-1} \frac{d\Sigma}{\mathcal{L}ds} \Sigma^{-1} = \frac{d}{\mathcal{L}ds} \left(\Sigma^{-1} \frac{d\Sigma}{\mathcal{L}ds} \Sigma^{-1} \right) \tag{26}$$

Second, we make the change of variable $dt = \mathcal{L}ds$ and we obtain

$$-\Sigma^{-1} \frac{d\Sigma}{dt} \Sigma^{-1} \frac{d\Sigma}{dt} \Sigma^{-1} = \frac{d}{dt} \left(\Sigma^{-1} \frac{d\Sigma}{dt} \Sigma^{-1} \right) \tag{27}$$

Here we slightly abuse notation since Σ is now a function of t instead of s. Note that the Lagrangian $\mathcal{L}$ depends on s through Σ and $\dot{\Sigma}$, therefore, t is a nonlinear function of s in general. Third, we use the identity $d\left(\Sigma^{-1}\right) = -\Sigma^{-1}(d\Sigma)\Sigma^{-1}$, and we multiply both sides of equation 27 by Σ. We arrive at

$$\frac{d^2\Sigma}{dt^2} = \frac{d\Sigma}{dt} \Sigma^{-1} \frac{d\Sigma}{dt} \tag{28}$$

The shortest path can be found by solving this second-order differential equation with boundary conditions $\Sigma(t_0) = \Sigma_0$ and $\Sigma(t_1) = \Sigma_1$.

The problem can be further simplified by noting that equation 28 is invariant for congruent transformations

$$\Sigma(t) = F\Sigma'(t)F^{\dagger} \tag{29}$$

where F is any non-singular matrix, which does *not* have to be orthogonal. Remarkably, we can choose the matrix F in a way that $\Sigma'(t)$ is diagonal along the entire path, from beginning to end. As proved by Theorem 7.6.4 in (Horn & Johnson, 2013), given two positive definite matrices Σ_0 and Σ_1, there is a non-singular matrix F and diagonal matrix D, such that

$$\Sigma_0 = FDF^\dagger \tag{30}$$

$$\Sigma_1 = FF^\dagger \tag{31}$$

The matrix F is given by $F = \Sigma_1^{1/2}U$, where the columns of U are the orthogonal eigenvectors of the matrix $\Sigma_1^{-1/2}\Sigma_0\Sigma_1^{-1/2}$, and D is the diagonal matrix of its eigenvalues. It is straightforward to verify that $FF^\dagger = \Sigma_1^{1/2}UU^\dagger\Sigma_1^{1/2} = \Sigma_1$. Furthermore, we have that $FDF^\dagger = \Sigma_1^{1/2}UDU^\dagger\Sigma_1^{1/2} = \Sigma_1^{1/2}\Sigma_1^{-1/2}\Sigma_0\Sigma_1^{-1/2}\Sigma_1^{1/2} = \Sigma_0$.

Under the congruent transformation 29, the shortest path equation 28 remains invariant

$$\frac{d^2\Sigma'}{dt^2} = \frac{d\Sigma'}{dt}\Sigma'^{-1}\frac{d\Sigma'}{dt} \tag{32}$$

However, boundary conditions are now diagonal, namely

$$\Sigma'(t_0) = D \tag{33}$$

$$\Sigma'(t_1) = \mathrm{I} \tag{34}$$

therefore equation 32 reduces to a set of independent scalar equations, one equation for each term in the diagonal of Σ'. The solution is equal to

$$\Sigma'(t) = D^{(t_1-t)/(t_1-t_0)} \tag{35}$$

Note that the exponent $(t_1 - t)/(t_1 - t_0)$ measures precisely the relative distance travelled along the path. In fact, since by definition $dt = \mathcal{L}ds$, then $t_1 - t_0$ measures the total length $\mathcal{D}$

$$t_1 - t_0 = \int_{s_0}^{s_1} ds\ \mathcal{L}(\Sigma(s), \dot{\Sigma}(s)) = \mathcal{D} \tag{36}$$

We do not compute the value of $\mathcal{D}$, instead we rescale t and use it to measure the *relative* distance travelled along the path. Therefore, we rewrite 35 as $\Sigma'(t) = D^{1-t}$ and the solution for Σ is equal to

$$\Sigma(t) = FD^{1-t}F^\dagger = \Sigma_1^{1/2}UD^{1-t}U^\dagger\Sigma_1^{1/2} = \tag{37}$$

$$= \Sigma_1^{1/2}\left(\Sigma_1^{-1/2}\Sigma_0\Sigma_1^{-1/2}\right)^{1-t}\Sigma_1^{1/2} \tag{38}$$

□

B. Forward process

Theorem B.1. *Given a random vector $\mathbf{x}_0$ of zero mean and covariance Σ_0, and another random vector $\boldsymbol{\epsilon}_t$ of zero mean and covariance I (isotropic), where $\mathbf{x}_0$ and $\boldsymbol{\epsilon}_t$ are uncorrelated. Assume the matrix $(I - \Sigma_0)$ is invertible. Define the matrix $\Phi_t = (I - \Sigma_0^{1-t})(I - \Sigma_0)^{-1}$ and note that, for $t \in (0, 1)$, Φ_t and $(\mathrm{I} - \Phi_t)$ are positive definite and, respectively, monotonically decreasing and increasing functions of Σ_0. Then, the corrupted vector $\mathbf{x}_t$ defined by*

$$\mathbf{x}_t = \Phi_t^{\frac{1}{2}}\mathbf{x}_0 + (I - \Phi_t)^{\frac{1}{2}}\boldsymbol{\epsilon}_t \tag{39}$$

has zero mean and covariance equal to Σ_0^{1-t}.

Proof. The mean of $\mathbf{x}_t$ is zero, because the mean of both $\mathbf{x}_0$ and $\boldsymbol{\epsilon}_t$ are zero. The covariance of $\mathbf{x}_t$ can be calculated from equation 39 by computing $\mathbf{x}_t\mathbf{x}_t^\dagger$ and averaging. Note that $\mathbf{x}_0$ and $\boldsymbol{\epsilon}_t$ are uncorrelated. Then, the covariance is equal to

$$\Sigma_t = \Phi_t^{\frac{1}{2}}\Sigma_0\Phi_t^{\frac{1}{2}} + (\mathrm{I} - \Phi_t) \tag{40}$$

Using the expression of $\Phi_t = (\mathrm{I} - \Sigma_0^{1-t})(\mathrm{I} - \Sigma_0)^{-1}$, we note that Φ_t and Σ_0 commute, therefore the covariance can be rewritten as

$$\Sigma_t = \Phi_t\Sigma_0 + (\mathrm{I} - \Phi_t) = \mathrm{I} - \Phi_t(\mathrm{I} - \Sigma_0) = \Sigma_0^{1-t} \tag{41}$$

□

C. Time complexity of power spectrum

The time complexity of DFT and fit of the power spectrum are, respectively, quasilinear ($O(d\log(d))$) and linear ($O(d)$) in the number of pixels d. In our experiments, it took a few minutes to obtain the constants c_1 and c_2 for CIFAR. We also computed the power spectrum of ImageNet 64x64 on a CPU in about one hour. In both cases, the time required to calculate these hyperparameters is much less than the time required for training. Because the complexity of the power spectrum is not worse than the complexity of training, we expect the computation time of the former to be always much smaller even for datasets with higher resolution.

Concerning the dataset size n, the complexity of DFT scales linearly with n, while the complexity of the fit of the power spectrum does not depend on the dataset size. Therefore, we do not expect this to be the limiting factor for our method. Furthermore, for much larger datasets, an estimate of the power spectrum may be obtained from a subset of the data.

D. Generated images

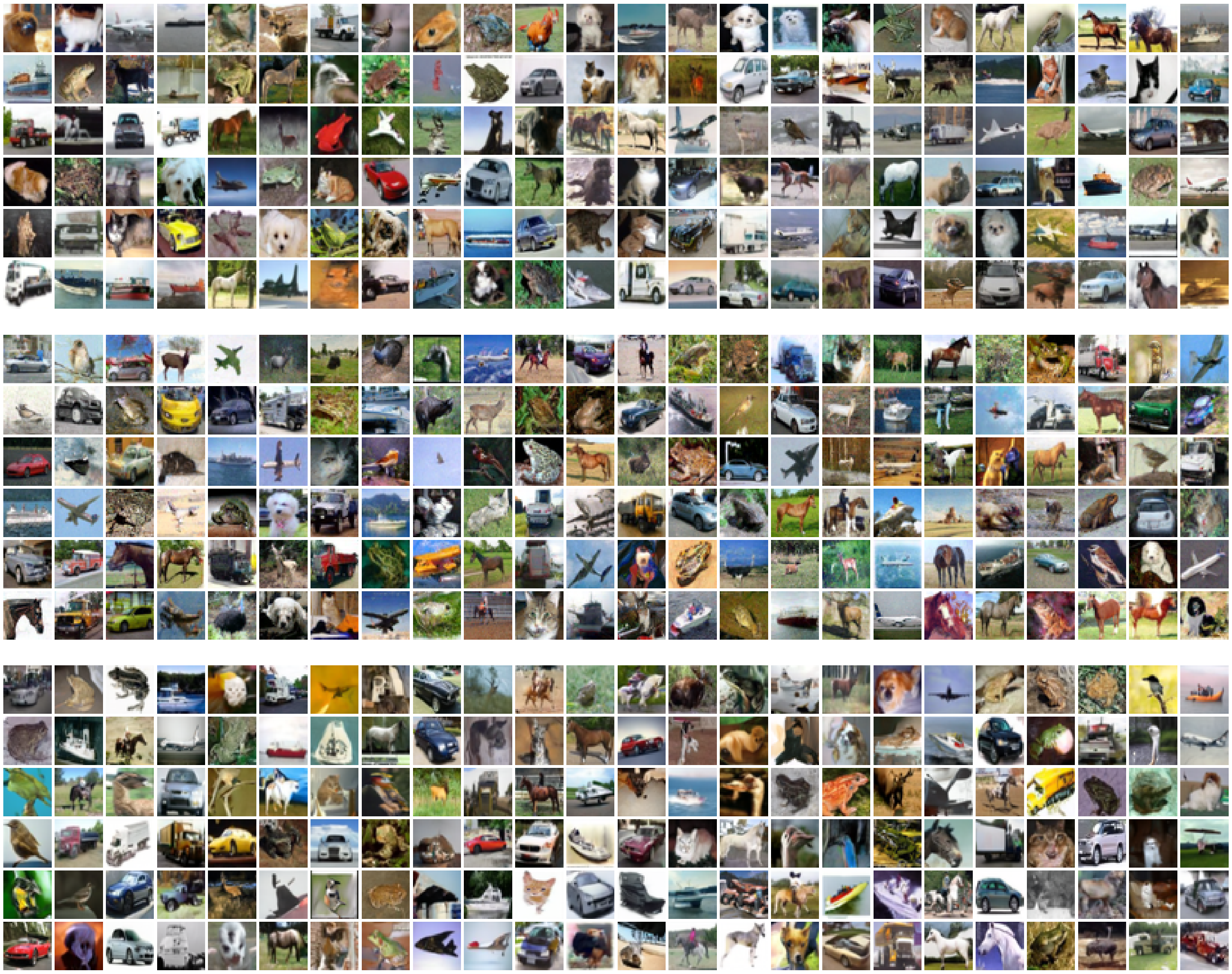

Figure 7. **Example generated images at T=500 diffusion steps for models trained on CIFAR10**. Top to bottom: SPD (FID = 2.74), iDDPM (Nichol & Dhariwal, 2021a) (FID = 4.20) and iDDPM+DDIM (Song et al., 2021) (FID = 3.74).

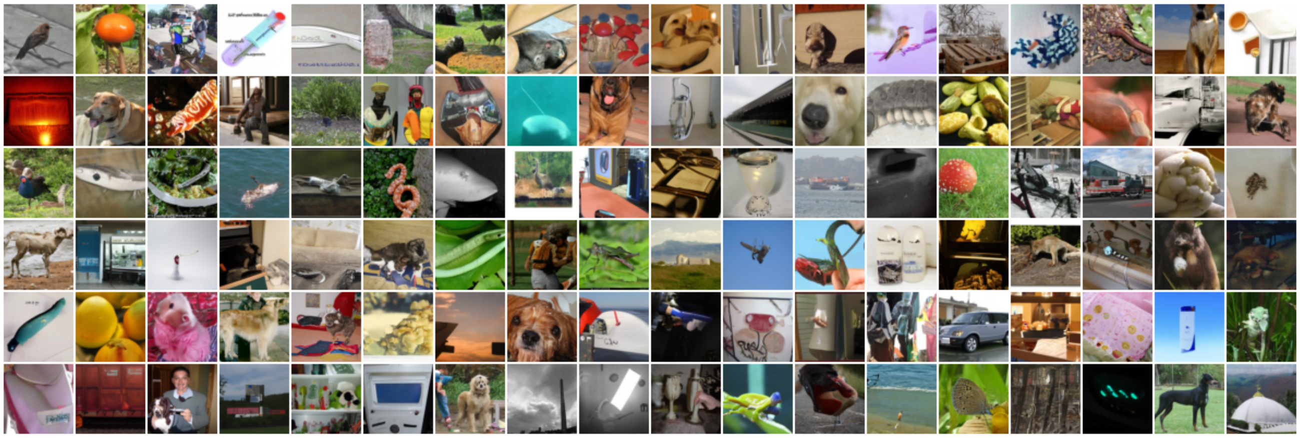

Figure 8. **Example generated images for model trained on ImagNet 64x64**. The model is trained at T=1000 diffusion steps and the evaluated FID is 13.7.