Subspace Diffusion Generative Models Supplementary Material

Bowen Jing*, Gabriele Corso*, Renato Berlinghieri, and Tommi Jaakkola

Massachusetts Institute of Technology
{bjing, gcorso, renb}@mit.edu, tommi@csail.mit.edu

1 Synthetic experiment

In order to validate the findings obtained in image generation on more generic data, we test the *subspace diffusion* framework on a synthetic dataset in $\mathbb{R}^{30}$. The dataset is a mixture of 100 Gaussians, each with isotropic variance $\sigma^2 = 0.05^2$, and whose centers are first sampled from a unit Gaussian and then modified such that the total variance is 50% and 75% explained by 6 and 11 PCA components, respectively. 64000 samples are drawn from the mixture of Gaussians to form the training dataset and are diffused under the variance exploding SDE with $\sigma_{\min} = 0.01, \sigma_{\max} = 13$. We train simple 3-layer feedforward score models in each optimal subspace—that is, the subspaces spanned by principal components—of dimensions 1–29. With each model, we generate 6400 samples in conjunction with a full-dimensional model at varying transition times from 0–1 in increments of 0.01. We use the Euler-Maruyama solver with 100 steps and Langevin corrections with a signal-to-noise ratio of 0.2.

The sample quality is evaluated in terms of the mean L_2 distance from each sampled point to the nearest training point, shown below in Figure 1. A U-shaped trend in sample quality is again observed as the transition time is varied. All subspaces with dimension ≥ 7 improve over the full-dimensional model (bottom row, mean distance ≈ 5.4), reinforcing the observation made in the image generation experiments. These experiments also highlights the greater generality of subspace diffusion compared to techniques that are limited to image generation such as cascading diffusion models.

2 Patch-PCA

We investigate the optimality of the downsampling subspaces in comparison with the best possible subspaces that produce an image-structured latent. Recall that the downsampling subspaces are defined as follows: suppose we have a full-resolution image $\mathbf{X} \in \mathbb{R}^{(n \times n \times 3)}$, with n an integer power of 2. Then the downsampled image $\mathbf{X}' \in \mathbb{R}^{(n/2 \times n/2 \times 3)}$ satisfies

$$\mathbf{X}'[a, b] = \frac{1}{2} \sum_{(i,j) \in \{0,1\}^2} \mathbf{X}[2a + i, 2b + j] \tag{1}$$

* Equal contribution

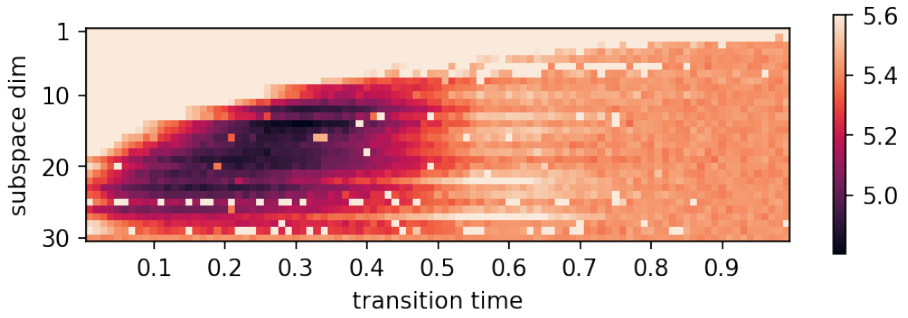

Fig. 1. Results on the synthetic dataset when varying the subspace dimension and transition time. The color indicates the sample quality in terms of the mean distance from each sampled point to the nearest training point (lower / darker is better). The results from the full-dimensional model alone are shown as the subspace model of dimension 30 in the bottom row and are (as expected) constant in quality.

where each element $\mathbf{X}[i, j]$ is an RGB color in $\mathbb{R}^3$. For output pixel $\mathbf{X}'[a, b]$, this is an operation over the 2×2 patch of pixels $\mathbf{X}[2a + i, 2b + j] \mid (i, j) \in \{0, 1\}^2$, which can be regarded as an element of a 12-dimensional vector space. That is,

$$\mathbf{X}'[a, b] = f(\mathbf{X}[2a + i, 2b + j] \mid (i, j) \in \{0, 1\}^2) \quad f : \mathbb{R}^{12} \to \mathbb{R}^3 \tag{2}$$

for f independent of a, b. The downsampling subspace corresponds to taking twice the mean of the input patch, but we can generalize to arbitrary linear functions and consider (2) with any linear f to define an image-structured subspace. The key aspect of this definition is that each basis element of the subspace corresponds to a *spatially localized* set of input features, and that the transformation operates identically for all spatial locations in the original image.

To find the optimal $n/2 \times n/2$ image-structured subspace we run PCA over the 12-dimensional distribution of patches $\mathbf{X}[2a + i, 2b + j] \mid (i, j) \in \{0, 1\}^2$ for all possible values of (a, b), and over all images (or as large a subset as is computationally feasible). We then project each patch onto the top three principal components to form the smaller image. This definition and procedure can be naturally extended to smaller subspaces by considering the input patches of 4×4, 8×8 pixels as vector spaces of dimensionality 48, 192, etc.

3 Hyperparameters

As mentioned in the main text, we did not tune any hyperparameters for training and directly used the default settings from the full-dimensional model, including checkpoint intervals. The sole exception was that we used reduced batch sizes due to different hardware constraints. We report FID and IS for the SDE sampler on CIFAR-10 using the best training checkpoint, as in previous work. All other results are obtained using the last training checkpoint.

During inference, the only hyperparameter tuned was the number of conditional Langevin steps. We tried 0, 1, 2, 5, or 10 steps using the last training checkpoint of the 8×8 NCSN++ model and chose the value leading to the best FID averaged across the cutoff times. We then used 2 steps for all experiments with the SDE sampler. The Langevin signal-to-noise ratio was fixed to 0.22 for NCSN++ and 0.01 for DDPM++ based on the best settings found in previous work. All other inference hyperparameters were fixed to their default values.

4 Detailed results

Model	Subspace	Threshold	t_1	Runtime	FID ↓	IS ↑
NCSN++ shallow (VE)	None	–	–	100%	2.38	9.93
	$16 \to 32$	1×10^{-4}	0.64	75%	2.45	9.81
		3×10^{-4}	0.60	72%	2.37	9.87
		1×10^{-3}	0.56	69%	2.31	9.95
		3×10^{-3}	0.52	66%	**2.29**	**9.99**
		1×10^{-2}	0.47	63%	2.46	9.96
		3×10^{-2}	0.42	59%	2.67	9.93
	$8 \to 32$	1×10^{-4}	0.69	72%	2.41	9.90
		3×10^{-4}	0.64	68%	2.39	9.83
		1×10^{-3}	0.60	64%	**2.29**	9.92
		3×10^{-3}	0.56	60%	2.35	10.08
		1×10^{-2}	0.51	56%	2.74	10.09
		3×10^{-2}	0.46	52%	3.42	**10.10**
NCSN++ deep (VE)	None	–	–	100%	2.20	9.89
	$16 \to 32$	1×10^{-4}	0.64	75%	2.25	9.86
		3×10^{-4}	0.60	73%	2.19	9.93
		1×10^{-3}	0.56	69%	**2.17**	**9.94**
		3×10^{-3}	0.52	67%	2.23	9.91
		1×10^{-2}	0.47	63%	2.31	9.90
		3×10^{-2}	0.42	60%	2.51	9.82
	$8 \to 32$	1×10^{-4}	0.69	72%	2.22	9.85
		3×10^{-4}	0.64	68%	2.24	9.87
		1×10^{-3}	0.60	64%	**2.21**	9.92
		3×10^{-3}	0.56	60%	2.39	**10.08**
		1×10^{-2}	0.51	56%	3.05	10.01
		3×10^{-2}	0.46	51%	3.51	9.96

Table 1. NCSN++ subspace diffusion results on the CIFAR-10 unconditional generation. Runtimes are reported as percentages of the respective full diffusion model.

Model	Subspace	Threshold	t_1	Runtime	FID ↓	IS ↑
DDPM++ shallow (sub-VP)	None	–	–	100%	2.61	9.56
	$16 \to 32$	1×10^{-4}	0.56	69%	**2.61**	9.53
		3×10^{-4}	0.51	65%	2.63	9.64
		1×10^{-3}	0.45	61%	2.75	**9.66**
		3×10^{-3}	0.39	56%	3.11	9.53
		1×10^{-2}	0.32	52%	4.07	9.52
		3×10^{-2}	0.26	47%	5.68	9.37
	$8 \to 32$	1×10^{-4}	0.62	65%	**2.60**	9.54
		3×10^{-4}	0.57	60%	2.68	9.56
		1×10^{-3}	0.50	55%	2.93	**9.66**
		3×10^{-3}	0.45	50%	3.73	9.63
		1×10^{-2}	0.38	43%	5.24	9.51
		3×10^{-2}	0.31	37%	7.61	9.23
DDPM++ deep (sub-VP)	None	–	–	100%	2.41	9.57
	$16 \to 32$	1×10^{-4}	0.56	69%	**2.40**	9.66
		3×10^{-4}	0.50	66%	2.43	9.62
		1×10^{-3}	0.44	61%	2.55	9.65
		3×10^{-3}	0.38	57%	2.84	**9.68**
		1×10^{-2}	0.32	53%	3.49	9.55
		3×10^{-2}	0.26	48%	4.64	9.52
	$8 \to 32$	1×10^{-4}	0.62	65%	**2.46**	9.67
		3×10^{-4}	0.56	60%	2.52	9.67
		1×10^{-3}	0.50	55%	2.76	**9.72**
		3×10^{-3}	0.44	50%	3.41	9.65
		1×10^{-2}	0.38	43%	4.39	9.55
		3×10^{-2}	0.31	37%	6.32	9.30

Table 2. DDPM++ subspace diffusion results on the CIFAR-10 unconditional generation. Runtimes are reported as percentages of the respective full diffusion model.

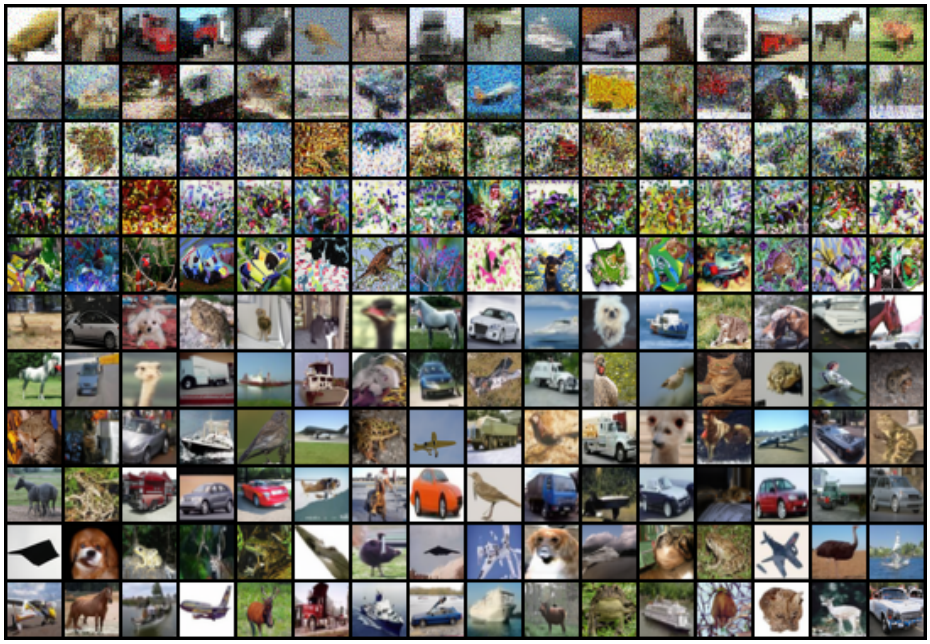

Fig. 2. Random samples from CIFAR-10 using the NCSN++ deep 16x16 subspace diffusion, each row represents samples with an extra 10% of the diffusion on the full-dimensional space (from 0% at the top to 100% at the bottom). High quality samples start appearing between 50-60% of full diffusion.

Fig. 3. Random samples from CelebA-HQ using the NCSN++ 64x64 subspace diffusion, each row represents samples with an extra 10% of the diffusion on the full-dimensional space (from 0% at the top to 100% at the bottom). High quality samples start appearing between 30-40% of full diffusion.

Fig. 4. Random samples from LSUN Church using the NCSN++ 64x64 subspace diffusion, each row represents samples with an extra 10% of the diffusion on the full-dimensional space (from 0% at the top to 100% at the bottom). High quality samples start at around 40% of full diffusion.

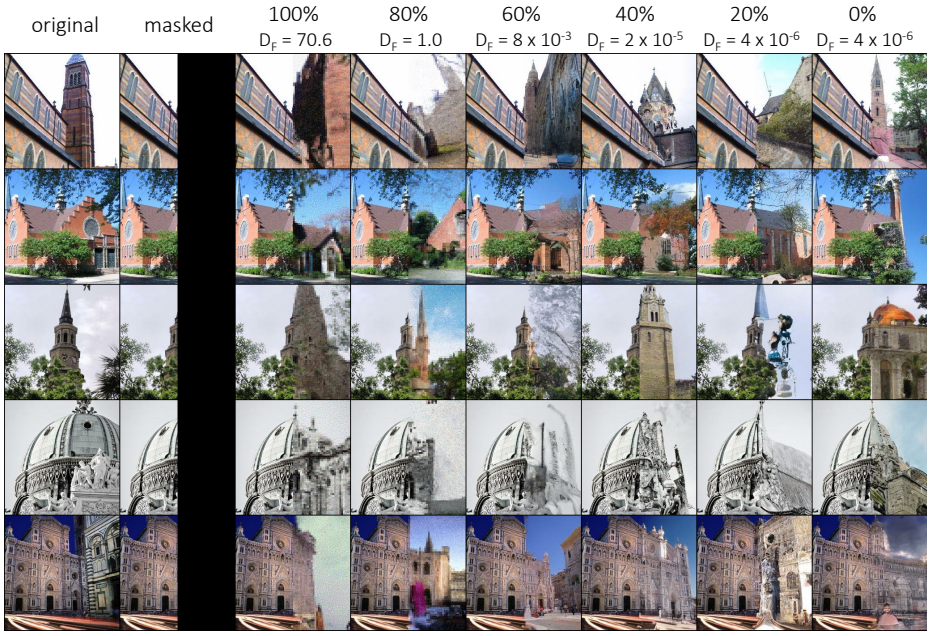

Fig. 5. Random samples of inpainting procedure from LSUN Church using the NCSN++ 64x64 subspace diffusion, with different proportions of subspace diffusion (reported at the top along with the corresponding Fisher divergence threshold).