A THEORETICAL ANALYSIS

Proposition A.1. *Let θ^* be optimal, i.e. $\theta^* = \arg\min_\theta L_{MSE}$. Then it satisfies*

$$D_{\theta^*}(\tilde{x}) = \tilde{x} + \sigma^2 \nabla_{\tilde{x}} \log p(\tilde{x}), \tag{1}$$

where $\tilde{x} \sim \mathcal{N}(x, \sigma^2 I)$.

Proof. Assume data density $p(x)$ be differentiable, then the optimal denoisier, i.e.

$$D^* \in \arg\min_D \mathbb{E}_{x\sim p(x), \tilde{x}\sim p(\tilde{x}|x)}[\|D(\tilde{x}) - x\|_2^2]$$

is given by

$$D^*(\tilde{x}) = \mathbb{E}_{x\sim p(x|\tilde{x})}[x]. \tag{2}$$

First note that the smooth density $p_{\sigma^2}(x)$ is given by

$$p_{\sigma^2}(\tilde{x}) = \int p(x, \tilde{x}) dx \tag{3}$$

$$= \int p(\tilde{x}|x) p(x) dx \tag{4}$$

where $p(\tilde{x}|x) = \mathcal{N}(x, \sigma^2 I)$. Then the gradient of smooth density is

$$\nabla p_{\sigma^2}(\tilde{x}) = \int \nabla p(\tilde{x}|x) p(x) dx \tag{5}$$

$$= \int \frac{(x - \tilde{x})}{\sigma^2} p(\tilde{x}|x) p(x) dx \tag{6}$$

$$= \frac{1}{\sigma^2} \int (x - \tilde{x}) p(x|\tilde{x}) p_{\sigma^2}(\tilde{x}) dx \tag{7}$$

$$= \frac{p_{\sigma^2}(\tilde{x})}{\sigma^2} \left(\int x p(x|\tilde{x}) dx - \tilde{x} \int p(x|\tilde{x}) dx \right) \tag{8}$$

$$= \frac{p_{\sigma^2}(\tilde{x})}{\sigma^2} \left(\mathbb{E}_{p(x|\tilde{x})}[x] - \tilde{x} \right) \tag{9}$$

$$= \frac{p_{\sigma^2}(\tilde{x})}{\sigma^2} \left(D^*(\tilde{x}) - \tilde{x} \right) \tag{10}$$

which results in

$$D(\tilde{x}) = \tilde{x} + \frac{\sigma^2}{p_{\sigma^2}(\tilde{x})} \nabla p_{\sigma^2}(\tilde{x}) = \tilde{x} + \sigma^2 \nabla \log p_{\sigma^2}(\tilde{x}) \tag{11}$$

□

Before we prove theorem 3.2, we introduce several lemmas.

Definition A.1. (Strong convexity). A function $f : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex if, for all $\mathbf{x}_1, \mathbf{x}_2$, the following inequality holds for some $\mu > 0$:

$$f(\mathbf{x}_2) \geq f(\mathbf{x}_1) + \nabla f(\mathbf{x}_1)^\top (\mathbf{x}_2 - \mathbf{x}_1) + \frac{\mu}{2} \|\mathbf{x}_1 - \mathbf{x}_2\|^2 \tag{12}$$

Definition A.2. (Strong log-concavity). A distribution $p : \mathbb{R}^d \to [0, 1]$ is Σ-strongly log-concave if, p of the form

$$p(\mathbf{x}) = g(\mathbf{x}) \mathcal{N}(0, \Sigma) \tag{13}$$

for some log-concave function g and a positive definite matrix Σ. If $\Sigma = \sigma^2 I$, p is σ^2-strongly log-concave shortly.

Following lemma shows the relationship between strong log-concavity and strong convexity.

Lemma A.2. *Assume p be σ^2-strongly log-concave, then $p \propto \exp(-f)$ for some $\frac{1}{\sigma^2}$-strongly convex f.*

The proof can be found in . Next lemma states the preservation of strong log-concavity under convolution.

Lemma A.3. *If p_1 is σ_1^2-strongly log-concave, and p_2 is σ_2^2-strongly concave, then the distribution $p_1 * p_2$ is $(\sigma_1^2 + \sigma_2^2)$-strongly log-concave.*

The proof can be found in . Finally, we have following lemma for the bounds on Wasserstein distance between p and its smoothed density p_{σ^2}.

Lemma A.4. *Let p be any distribution and p_{σ^2} be smoothed density obtained by $p_{\sigma^2} = p * \mathcal{N}(0, \sigma^2 I)$, then the 2-Wasserstein distance between p and p_{σ^2} satisfies*

$$\mathcal{W}_2(p, p_{\sigma^2}) \leq \sigma\sqrt{d} \tag{14}$$

Now we're ready to proof our theorem.

Proof. Let $\tilde{\mathbf{x}}$ be local optimum of $p_{\tilde{\sigma}^2}$. By lemma A.4, as $-\log p$ is μ-strongly convex, p is $\frac{1}{\mu}$-strongly log-concave and as Gaussian distribution $\mathcal{N}(0, \tilde{\sigma}^2)$ is $\tilde{\sigma}^2$-strongly log-concave, $p_{\tilde{\sigma}^2}$ is $(\frac{1}{\mu} + \tilde{\sigma}^2)$-strongly log-concave, and equivalently $-\log p_{\tilde{\sigma}^2}$ is $\frac{\mu}{1+\mu\tilde{\sigma}^2}$-strongly convex. Then as $\hat{\mathbf{x}} \in \arg\min L_{\text{MAP},\tilde{\sigma}}$, we have

$$\nabla L_{\text{MAP},\tilde{\sigma}}(\hat{\mathbf{x}}) = 0 \iff -\nabla p_{\tilde{\sigma}^2}(\hat{\mathbf{x}}) + \frac{1}{\sigma^2}(\hat{\mathbf{x}} - \mathbf{y}) = 0 \tag{15}$$

$$\iff -\nabla p_{\tilde{\sigma}^2}(\hat{\mathbf{x}}) + \nabla p_{\tilde{\sigma}^2}(\tilde{\mathbf{x}}) = \frac{1}{\sigma^2}(\mathbf{y} - \hat{\mathbf{x}}) \tag{16}$$

$$\iff \langle -\nabla p_{\tilde{\sigma}^2}(\hat{\mathbf{x}}) + \nabla p_{\tilde{\sigma}^2}(\tilde{\mathbf{x}}), \hat{\mathbf{x}} - \tilde{\mathbf{x}} \rangle = \frac{1}{\sigma^2}\langle \mathbf{y} - \hat{\mathbf{x}}, \hat{\mathbf{x}} - \tilde{\mathbf{x}} \rangle \tag{17}$$

Then we have

$$\frac{1}{\sigma^2}\langle \mathbf{y} - \tilde{\mathbf{x}}, \hat{\mathbf{x}} - \tilde{\mathbf{x}} \rangle = \frac{1}{\sigma^2}\langle (\mathbf{y} - \hat{\mathbf{x}}) + (\hat{\mathbf{x}} - \tilde{\mathbf{x}}), \hat{\mathbf{x}} - \tilde{\mathbf{x}} \rangle \tag{18}$$

$$= \frac{1}{\sigma^2}\langle \mathbf{y} - \hat{\mathbf{x}}, \hat{\mathbf{x}} - \tilde{\mathbf{x}} \rangle + \frac{1}{\sigma^2}\langle \hat{\mathbf{x}} - \tilde{\mathbf{x}}, \hat{\mathbf{x}} - \tilde{\mathbf{x}} \rangle \tag{19}$$

$$= \langle -\nabla p_{\tilde{\sigma}^2}(\hat{\mathbf{x}}) + \nabla p_{\tilde{\sigma}^2}(\tilde{\mathbf{x}}), \hat{\mathbf{x}} - \tilde{\mathbf{x}} \rangle + \frac{1}{\sigma^2}\|\hat{\mathbf{x}} - \tilde{\mathbf{x}}\|_2^2 \tag{20}$$

$$\geq \frac{\mu}{1 + \mu\tilde{\sigma}^2}\|\hat{\mathbf{x}} - \tilde{\mathbf{x}}\|_2^2 + \frac{1}{\sigma^2}\|\hat{\mathbf{x}} - \tilde{\mathbf{x}}\|_2^2 \tag{21}$$

$$= \frac{1 + \mu\tilde{\sigma}^2 + \mu\sigma^2}{\sigma^2(1 + \mu\tilde{\sigma}^2)}\|\hat{\mathbf{x}} - \tilde{\mathbf{x}}\|_2^2 \tag{22}$$

Then by Cauchy-Schwarz inequality, we have

$$\|\hat{\mathbf{x}} - \tilde{\mathbf{x}}\|_2 \leq \frac{1 + \mu\tilde{\sigma}^2}{1 + \mu\tilde{\sigma}^2 + \mu\sigma^2}\|\hat{\mathbf{x}} - \mathbf{y}\|_2 = \frac{1 + \mu\tilde{\sigma}^2}{1 + \mu\tilde{\sigma}^2 + \mu\sigma^2}\|\mathbf{u}\|_2 \tag{23}$$

Finally, by lemma A.5,

$$\mathbb{E}\|\hat{\mathbf{x}} - \mathbf{x}^*\|_2 \leq \mathbb{E}\|\hat{\mathbf{x}} - \tilde{\mathbf{x}}\|_2 + \mathbb{E}\|\tilde{\mathbf{x}} - \mathbf{x}^*\|_2 \tag{24}$$

$$\leq \frac{1 + \mu\tilde{\sigma}^2}{1 + \mu\tilde{\sigma}^2 + \mu\sigma^2}\mathbb{E}_{\mathbf{u}\sim\mathcal{N}(0,\sigma^2 I)}[\|\mathbf{u}\|_2] + \mathcal{W}_2(p, p_{\tilde{\sigma}^2}) \tag{25}$$

$$= \frac{\sigma\sqrt{d}(1 + \mu\tilde{\sigma}^2)}{1 + \mu\tilde{\sigma}^2 + \mu\sigma^2} + \tilde{\sigma}\sqrt{d} \tag{26}$$

□

B EXPERIMENT DETAILS

B.1 TRAINING SCORE NETWORKS

We used NCSNv2 from Song & Ermon (2020). For CIFAR-10 we used original NCSNv2, and for ImageNet we used the deepest version of NCSNv2. Note that the author first released NCSN (Song & Ermon, 2019), then proposed improved version (Song & Ermon, 2020). NCSN and NCSN v2 are based on RefineNet, and some major changes in normalization, pooling layer, and convolution layer lead to successful score-based modeling. The original NCSN was developed for generative modeling, and choosing noise level is crucial for generative modeling. Even though we are doing image denoising, choosing noise level also seems important. We experimented with two types of noise sequences: uniform sequence and geometric sequence. We used uniform noise sequence for one-step denoiser. We set $\sigma_1 = 1.0$ and $\sigma_L = 0.05$ with $L = 20$. We used geometric sequence for multi-step denoiser. For geometric noise sequences, we set $\sigma_1 = 1.0$, and $\sigma_L = 0.01$ with $L = 32$. Note that combining both sequences doesn't change the overall results.

For all experiments, we trained with Adam optimizer with learning rate 1e-5, and ran 300,000 iterations. We will soon release the code for details.

B.2 MULTI-STEP DENOISERS

For each Gaussian and uniform distribution, we ran annealed gradient descent with learning rate $\alpha = 2e-5$, and for laplace distribution we ran with learning rate $\alpha = 3e-5$. For each noise levels, we ran with $T = 1$ for fast denoising.

B.3 TRAINING CLASSIFIERS

For pretrained classifiers, we used CIFAR-10 classifiers publicly released from Salman et al. (2020), and pytorch pretrained ResNet50 for ImageNet. Also, for white-box smoothing baseline, we used Gaussian randomized smoothing baseline from Cohen et al. (2019) and ImageNet uniform and laplace ResNet50 baseline from Yang et al. (2020). Otherwise, we trained ResNet110 with laplace and uniform noise data augmentation on CIFAR-10 to reproduce the results. For training, we tested with $\sigma = \{0.15, 0.25, 0.50, 1.00\}$, with the training hyperparameter same as Cohen et al. (2019).

B.4 CERTIFICATION

We use the CERTIFY of randomized smoothing (Cohen et al., 2019) to do our experiments. We conducted all experiments with $n = 10,000$, $n0 = 100$ and $\alpha = 0.001$. Note that if we certify with larger n all results can be improved, however we stick with $n = 10,000$ due to computational constraints.

C Additional Experiments

C.1 How multi-scale methods help

In this section, we show how training with multi-scale DSM differs from training with each noise levels. To comparse, we trained score networks with one noise level each, and otherwise we trained with multi-scale DSM. We trained with noise levels $\sigma = 0.12, 0.25, 0.50, 1.00$, and plot certified accuracy for denoised smoothing with one-step denoiser from each score networks (Figure). that the multi-scale DSM achieves better performance, which is explained in section .

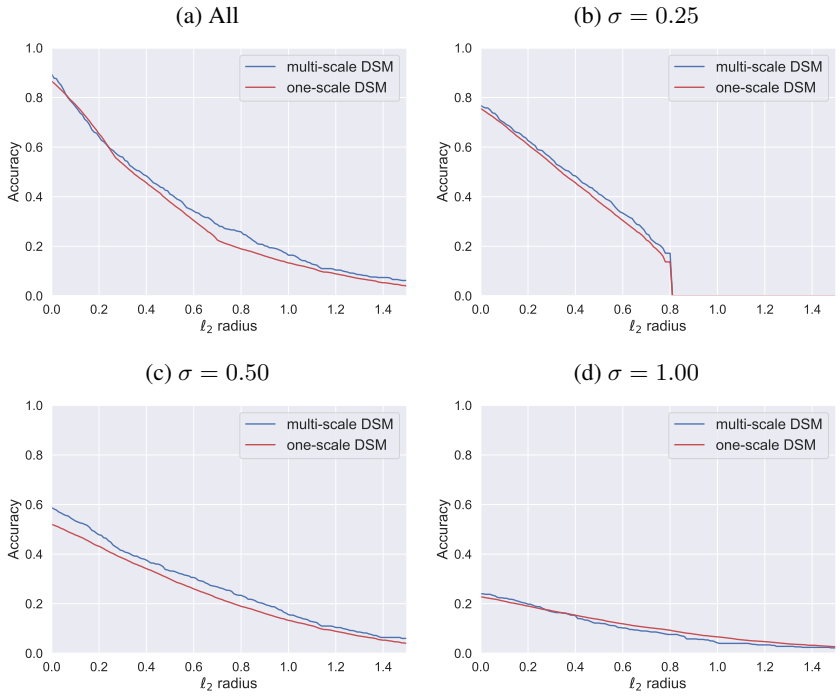


Figure 1: ddl

C.2 CERTIFICATION WITH OTHER CLASSIFIERS

Here we show that using stronger classifier, i.e. the classifier with high test accuracy, achieves better performance. We used 4 pretrained classifiers ResNet110, ResNet18, WideResNet40-10, WideResNet28-10 from Salman et al. (2020), where each classifier is trained with 300 epochs. We found out that using stronger classifier achieves better certified accuracy, and it is because we aren't fitting the denoiser into specific classifier (See Figure).

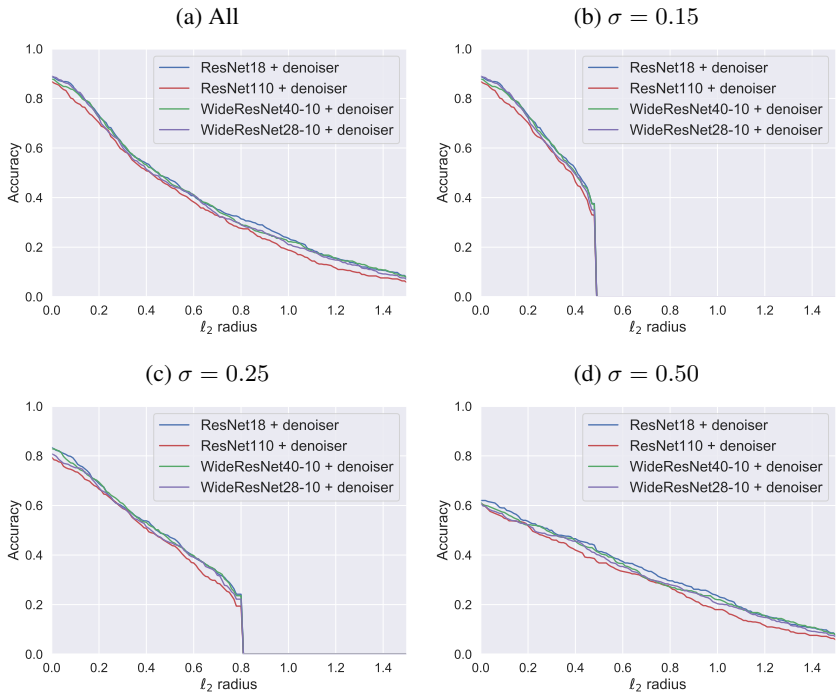
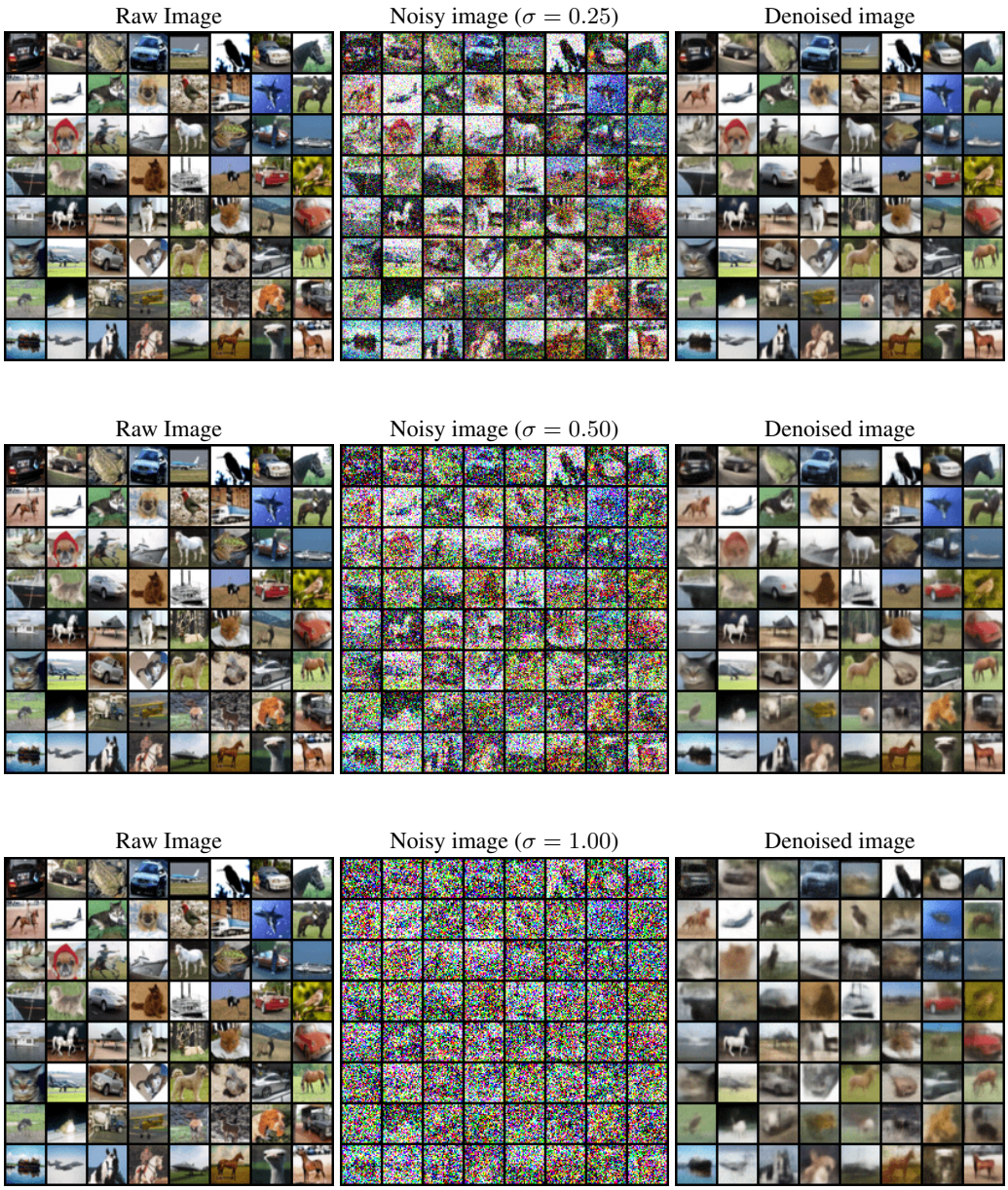


Figure 2: ddl

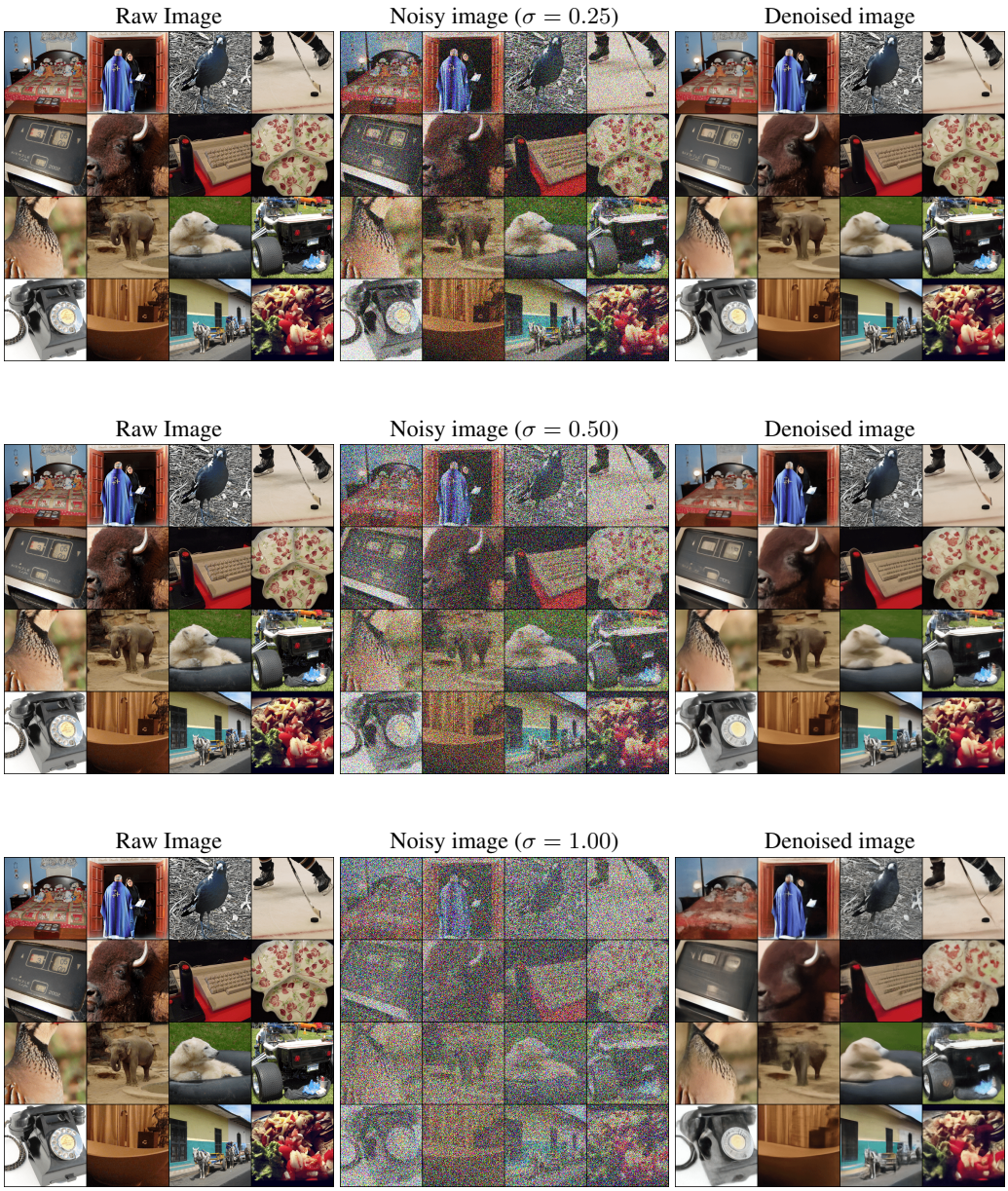
D DENOISED SAMPLES

D.1 ONE-STEP DENOISER

D.1.1 CIFAR-10, GAUSSIAN NOISE



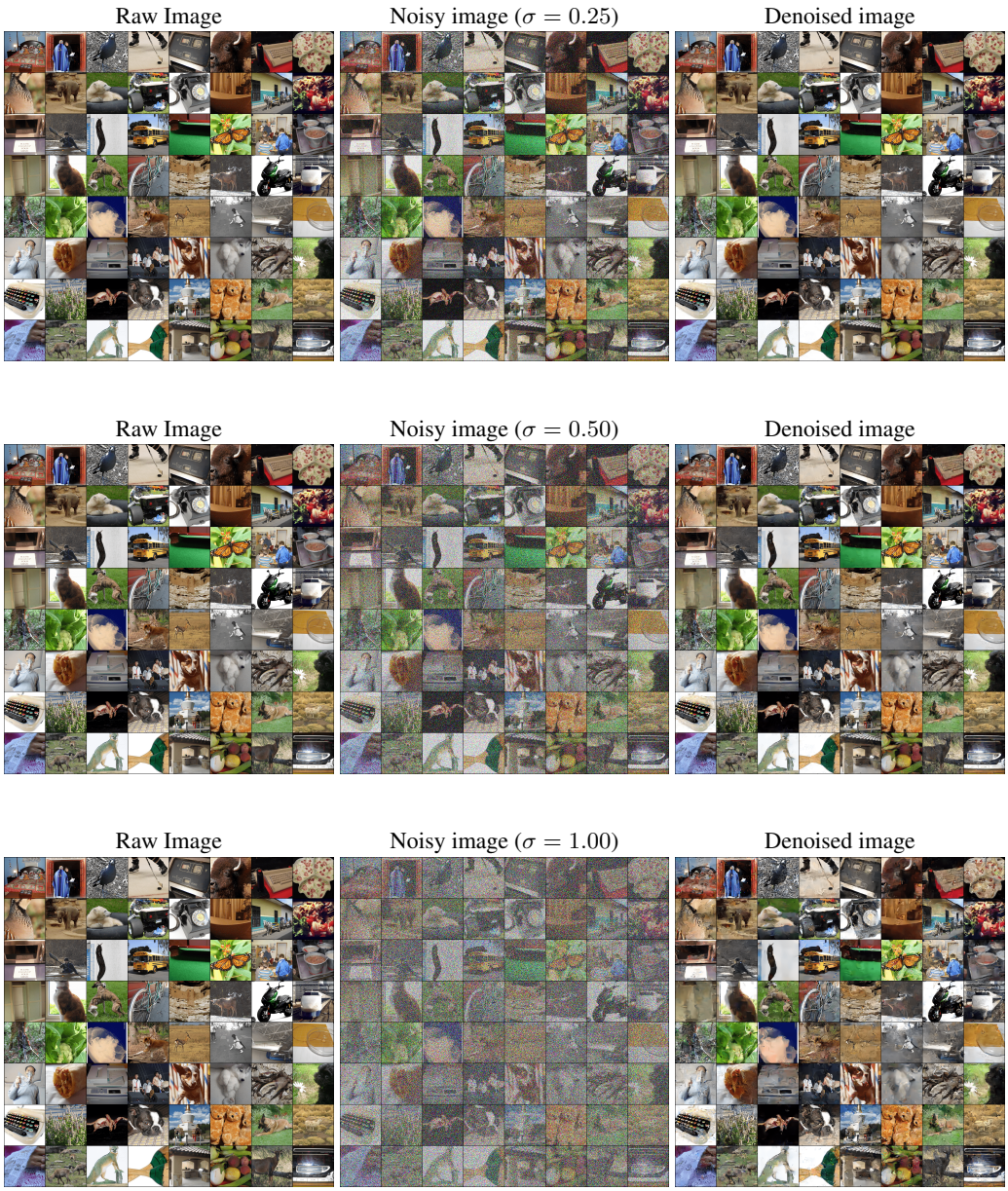
D.1.2 IMAGENET, GAUSSIAN NOISE



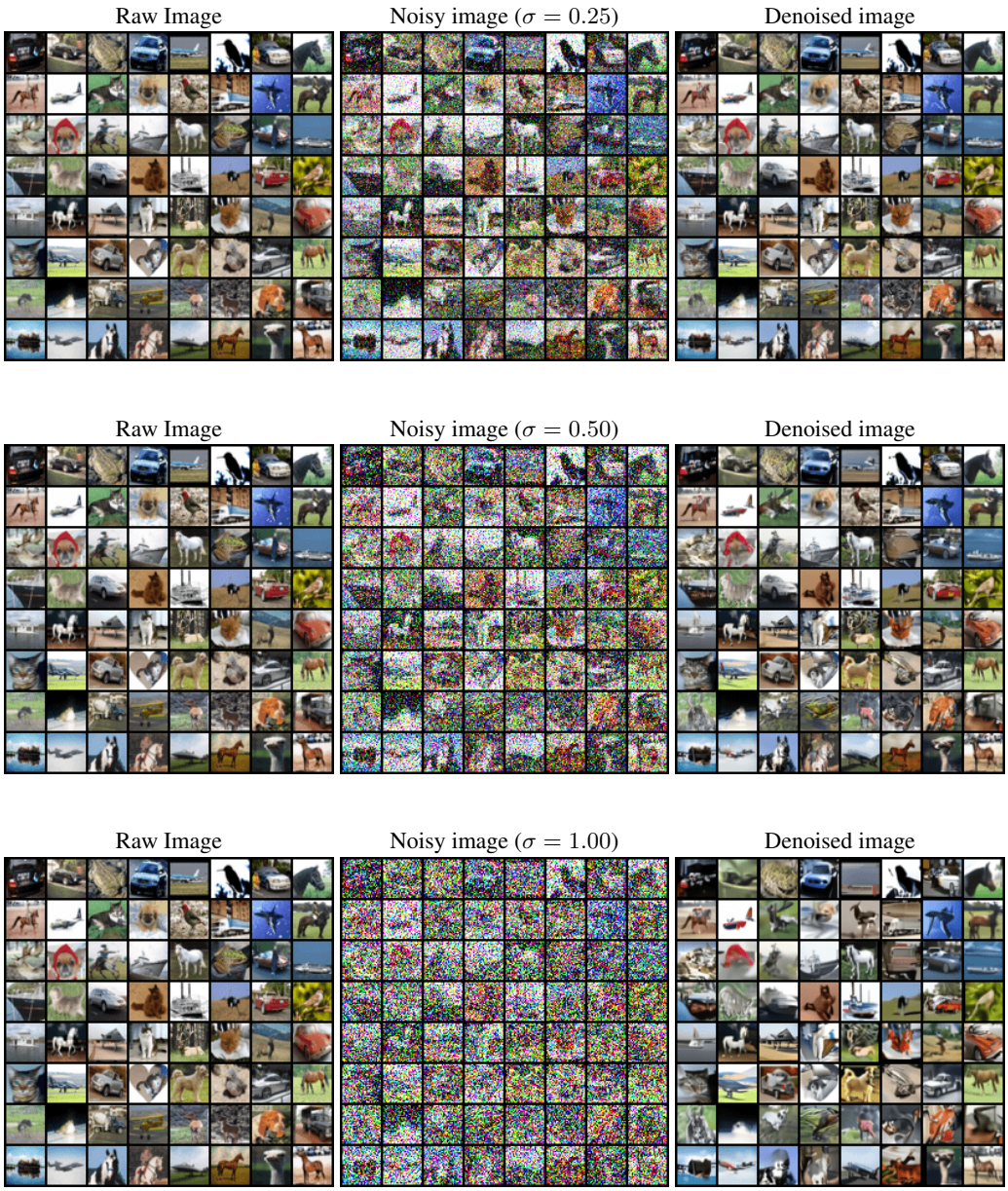
D.2 MULTI-STEP DENOISER

D.2.1 CIFAR-10, GAUSSIAN NOISE

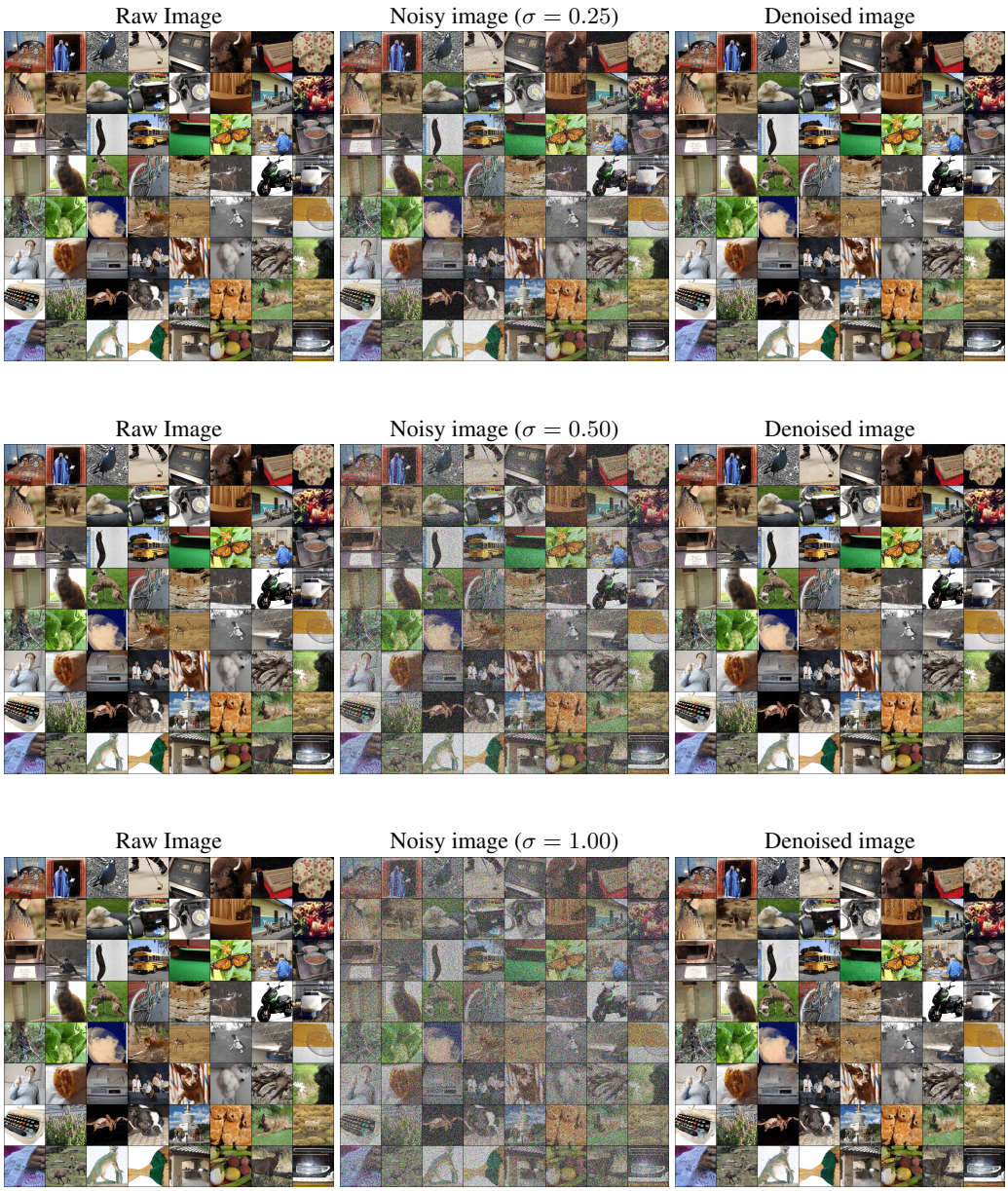
D.2.2 IMAGENET, GAUSSIAN NOISE



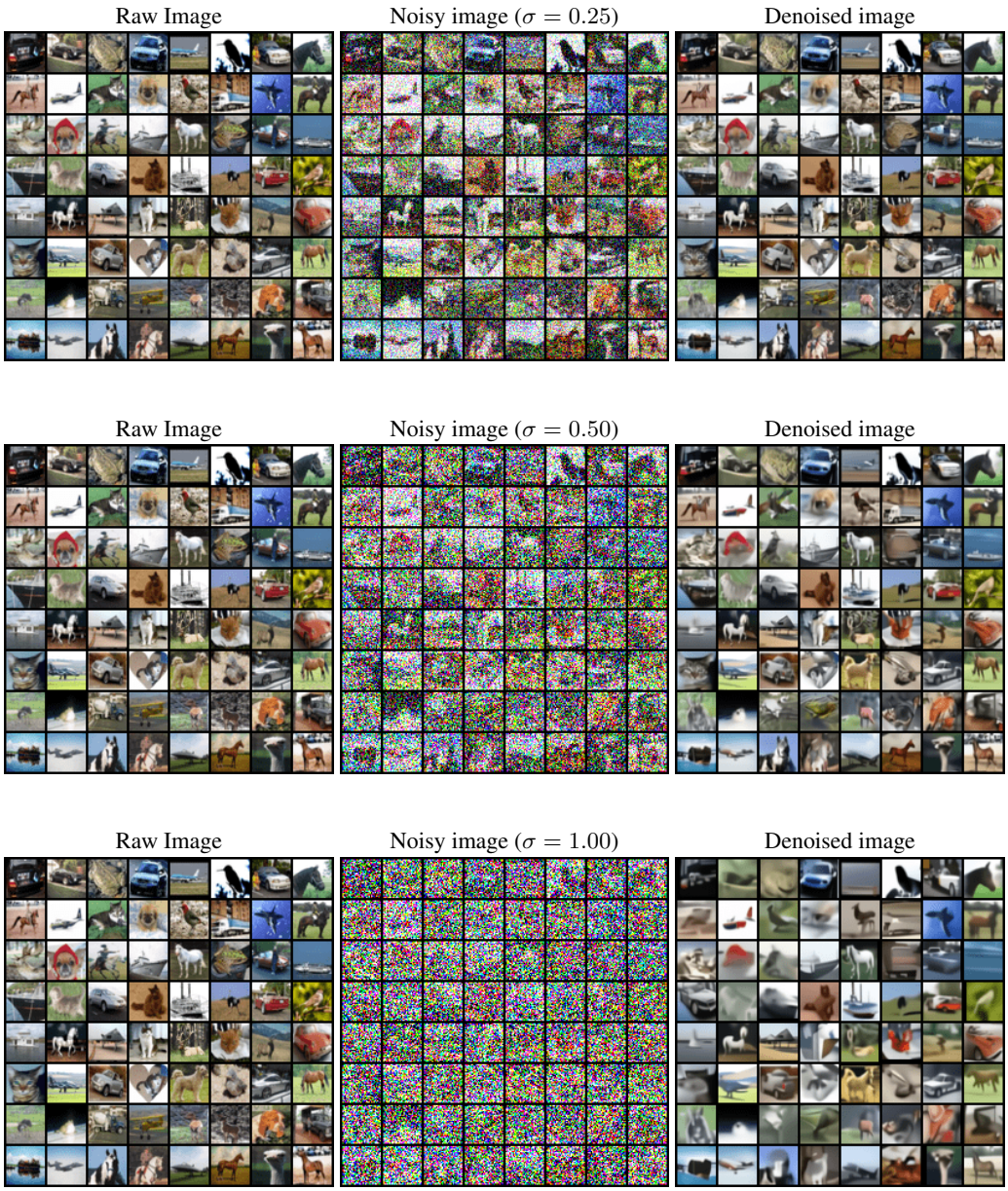
D.2.3 CIFAR-10, LAPLACE NOISE



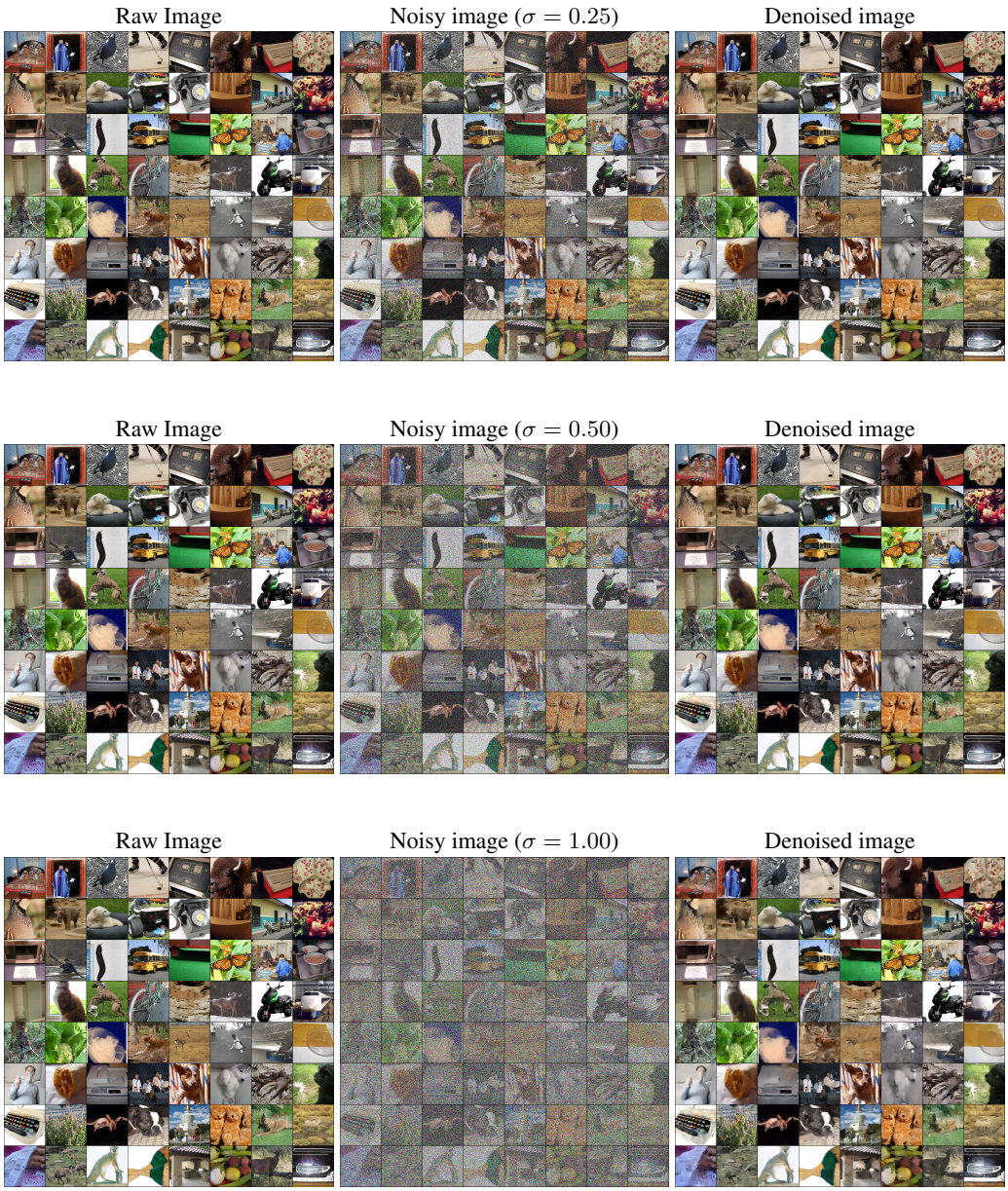
D.2.4 IMAGENET, LAPLACE NOISE



D.2.5 CIFAR-10, UNIFORM NOISE



D.2.6 IMAGENET, UNIFORM NOISE



REFERENCES

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pp. 1310–1320, 2019.

Hadi Salman, Mingjie Sun, Greg Yang, Ashish Kapoor, and J. Zico Kolter. Denoised smoothing: A provable defense for pretrained classifiers, 2020.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, pp. 11918–11930, 2019.

Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *arXiv preprint arXiv:2006.09011*, 2020.

Greg Yang, Tony Duan, Edward Hu, Hadi Salman, Ilya Razenshteyn, and Jerry Li. Randomized smoothing of all shapes and sizes. *arXiv preprint arXiv:2002.08118*, 2020.