A Scalable Training Strategy for Blind Multi-Distribution Noise Removal

Kevin Zhang, Sakshum Kulshrestha, and Christopher A. Metzler, *Member, IEEE*

***Abstract*— Despite recent advances, developing general-purpose universal denoising and artifact-removal networks remains largely an open problem: Given fixed network weights, one inherently trades-off specialization at one task (e.g., removing Poisson noise) for performance at another (e.g., removing speckle noise). In addition, training such a network is challenging due to the curse of dimensionality: As one increases the dimensions of the specification-space (i.e., the number of parameters needed to describe the noise distribution) the number of unique specifications one needs to train for grows exponentially. Uniformly sampling this space will result in a network that does well at very challenging problem specifications but poorly at easy problem specifications, where even large errors will have a small effect on the overall mean squared error. In this work we propose training denoising networks using an adaptive-sampling/active-learning strategy. Our work improves upon a recently proposed universal denoiser training strategy by extending these results to higher dimensions and by incorporating a polynomial approximation of the true specification-loss landscape. This approximation allows us to reduce training times by almost two orders of magnitude. We test our method on simulated joint Poisson-Gaussian-Speckle noise and demonstrate that with our proposed training strategy, a single blind, generalist denoiser network can achieve peak signal-to-noise ratios within a uniform bound of specialized denoiser networks across a large range of operating conditions. We also capture a small dataset of images with varying amounts of joint Poisson-Gaussian-Speckle noise and demonstrate that a universal denoiser trained using our adaptive-sampling strategy outperforms uniformly trained baselines.**

***Index Terms*— Denoising, active sampling, deep learning.**

I. Introduction

NEURAL networks have become the gold standard for solving a host of imaging inverse problems [1]. From denoising and deblurring to compressive sensing and phase retrieval, modern deep neural networks significantly outperform classical techniques like BM3D [2] and KSVD [3].

The most straightforward and common approach to apply deep learning to inverse problems is to train a neural network to learn a mapping from the space of corrupted images/measurements to the space of clean images. In this framework, one first captures or creates a training set consisting of clean images $x_1, x_2, \ldots$ and corrupted images $y_1, y_2, \ldots$ according to some known forward model $p(y_i|x_i, \theta)$, where $\theta \in \Theta$ denotes the latent variable(s) specifying the forward model. For example, when training a network to remove additive white Gaussian noise

$$p(y_i|x_i, \theta) = \frac{1}{\sigma\sqrt{2\pi}} \exp -\frac{\|y_i - x_i\|^2}{2\sigma^2}, \tag{1}$$

and the latent variable θ is the standard deviation σ. With a training set of L pairs $\{x_i, y_i\}_{i=1}^{L}$ in hand, one can then train a network to learn a mapping from y to x.

Typically, we are not interested in recovering signals from a single corruption distribution (e.g., a single fixed noise standard deviation σ) but rather a range of distributions. For example, we might want to remove additive white Gaussian noise with standard deviations anywhere in the range $[0, 50]$ ($\Theta = \{\sigma|\sigma \in [0, 50]\}$). The size of this range determines how much the network needs to generalize and there is inherently a trade-off between specialization and generalization. By and large, a network trained to reconstruct images over a large range of corruptions (a larger set Θ) will under-perform a network trained and specialized over a narrow range [4].

This problem becomes significantly more challenging when dealing with mixed, multi-distribution noise. As one increases the number of parameters (e.g., Gaussian standard deviation, Poisson rate, number of speckle realizations, ...) the space of corrupted signals one needs to reconstruct grows exponentially: The specification space becomes the Cartesian product (e.g., $\Theta = \Theta_{Gaussian} \times \Theta_{Poisson} \times \Theta_{speckle}$) of the spaces of each of the individual noise distributions.

This expansion does not directly prevent someone (with enough compute resources) from training a "universal" denoising algorithm. One can sample from Θ, generate a training batch, optimize the network to minimize some reconstruction loss, and repeat. However, this process depends heavily on the policy/probability-density-function π used to sample from Θ. As noted in [5] and corroborated in Section V, uniformly sampling from Θ will produce networks that do well on hard examples but poorly (relative to how well a specialized network performs) on easy examples.

A. Our Contribution

In this work, we develop an adaptive-sampling/active-learning strategy that allows us to train a single "universal" network to remove mixed Poisson-Gaussian-Speckle noise such that the network consistently performs within a uniform bound of specialized bias-free DnCNN baselines [4], [6]. Our

Received 30 October 2023; revised 11 June 2024 and 2 August 2024; accepted 28 September 2024. Date of publication 22 October 2024; date of current version 31 October 2024. This work was supported in part by the Air Force Office of Scientific Research (AFOSR) Young Investigator Program under Award FA9550-22-1-0208, in part by the Office of Naval Research (ONR) under Award N000142312752, in part by the National Science Foundation (NSF) CAREER under Award 2339616, and in part by a Seed Grant from SAAB Inc. The associate editor coordinating the review of this article and approving it for publication was Dr. Charles Kervrann. *(Corresponding author: Christopher A. Metzler.)*

Kevin Zhang and Christopher A. Metzler are with the Department of Computer Science, University of Maryland, College Park, MD 20742 USA (e-mail: metzler@umd.edu).

Sakshum Kulshrestha is with Waymo in Mountain View, CA 94043 USA.

Digital Object Identifier 10.1109/TIP.2024.3482185

key contribution is a novel, polynomial approximation of the specification-loss landscape. This approximation allows us to tractably apply (using over 50× fewer training examples than it would otherwise require) the adaptive-sampling strategy developed in [5], wherein training a denoiser is framed as a constrained optimization problem. We validate our technique with both simulated and experimentally captured data.

II. Related Work

Overcoming the specialization-generalization trade-off has been the focus of intense research efforts over the last 5 years.

A. Adaptive Denoising

One approach to improve generalization is to provide the network information about the current problem specifications θ at test time. For example, [7] demonstrated one could provide a constant standard-deviation map as an extra channel to a denoising network so that it could adapt to i.i.d. Gaussian noise. [8] extends this idea by adding a general standard-deviation map as an extra channel, to deal with spatially-varying Gaussian noise. This idea was recently extended to deal with correlated Gaussian noise [9]. The same framework can be extended to more complex tasks like compressive sensing, deblurring, and descattering as well [10], [11]. These techniques are all non-blind and require an accurate estimate of the specification parameters θ to be effective.

B. Universal Denoising

Somewhat surprisingly, the aforementioned machinery may be unnecessary if the goal is to simply remove additive white Gaussian noise over a range of different standard deviations: [6] recently demonstrated one can achieve significant invariance to the noise level by simply removing biases from the network architecture. Reference [12] also achieves similar invariance to noise level by scaling the input images to the denoiser to match the distribution it was trained on. Alternatively, at a potentially large computational cost, one can apply iterative "plug and play" or diffusion models that allow one to denoise a signal contaminated with noise with parameters θ' using a denoiser/diffusion model trained for minimum mean squared error additive white Gaussian noise removal [13], [14], [15]. These plug and play methods are non-blind and require knowledge of the likelihood $p(y|x, \theta')$ at test time.

C. Training Strategies

Generalization can also be improved by modifying the training set [16]. In the context of image restoration problems like denoising, [17] propose updating the training data sampling distribution each epoch to sample the data that the neural network performed worse on during the prior epoch preferentially, in an ad-hoc way. In [5] the authors developed a principled adaptive training strategy by framing training a denoiser across many problem specifications as a minimax optimization problem. This strategy will be described in detail in Section IV.

D. Relationship to Existing Works

We go beyond [5] by incorporating a polynomial approximation of the specification-loss landscape. This approximation is the key to scaling the adaptive training methodology to high-dimensional latent parameter spaces. It allows us to efficiently train a blind image denoiser that can operate effectively across a large range of noise conditions.

III. Problem Formulation

A. Noise Model

This paper focuses on removing joint Poisson-Gaussian-Speckle noise using a single blind image denoising network. Such noise occurs whenever imaging scenes illuminated by a coherent (e.g., laser) source. In this context, photon/shot noise introduces Poisson noise, read noise introduces Gaussian noise, and the constructive and destructive interference caused by the coherent fields scattered off optically rough surfaces causes speckle noise [18].

The overall forward model can be described by

$$y_i = \frac{1}{\alpha}\text{Poisson}(\alpha(r \circ w_i)) + n_i, \tag{2}$$

where the additive noise n_i follows a Gaussian distribution $\mathcal{N}(0, \sigma^2\mathbf{I})$; the multiplicative noise w_i follows a Gamma distribution with concentration parameter B/β and rate parameter B/β, where B is the upper bound on β; and α is a scaling parameter than controls the amount of Poisson noise. The forward model is thus specified by the set of latent variables $\theta = \{\sigma, \alpha, \beta\}$.

A few example images generated according to this forward model are illustrated in Figure 1. Variations in the problem specifications results in drastically different forms of noise.

B. Specification-Loss Landscape

A *specification* is a set of n parameters that define a task. In our setting, the specifications are the distribution parameters describing the noise in an image. Each of these parameters is bounded in an interval $[l_i, r_i]$, for $1 \leq i \leq n$. The *specification space* Θ is the Cartesian product of these intervals: $\Theta = [l_1, r_1] \times \cdots \times [l_n, r_n]$. Suppose we have a function f that can solve a task (e.g., denoising) at any specification in Θ, albeit with some error. Then the *specification-loss landscape*, for a given f over Θ, is the function $\mathcal{L}_f$ which maps points θ from Θ to the corresponding error/loss (e.g., mean squared error) that f achieves at that specification.

Now suppose that all functions f under consideration come from some family of functions $\mathcal{F}$. Let the ideal function from $\mathcal{F}$ that solves a task at a particular specification θ be $f^\theta_{\text{ideal}} = \arg\min_{f\in\mathcal{F}} \mathcal{L}_f(\theta)$. With this in mind, we define the *ideal specification-loss landscape* as the function that maps points θ in Θ to the loss that f^θ_{ideal} achieves on the task with specification θ, and denote it $\mathcal{L}_{\text{ideal}}$.

C. The Uniform Gap Problem

Our goal is to find a single function $f^* \in \mathcal{F}$ that achieves consistent performance across the specification space

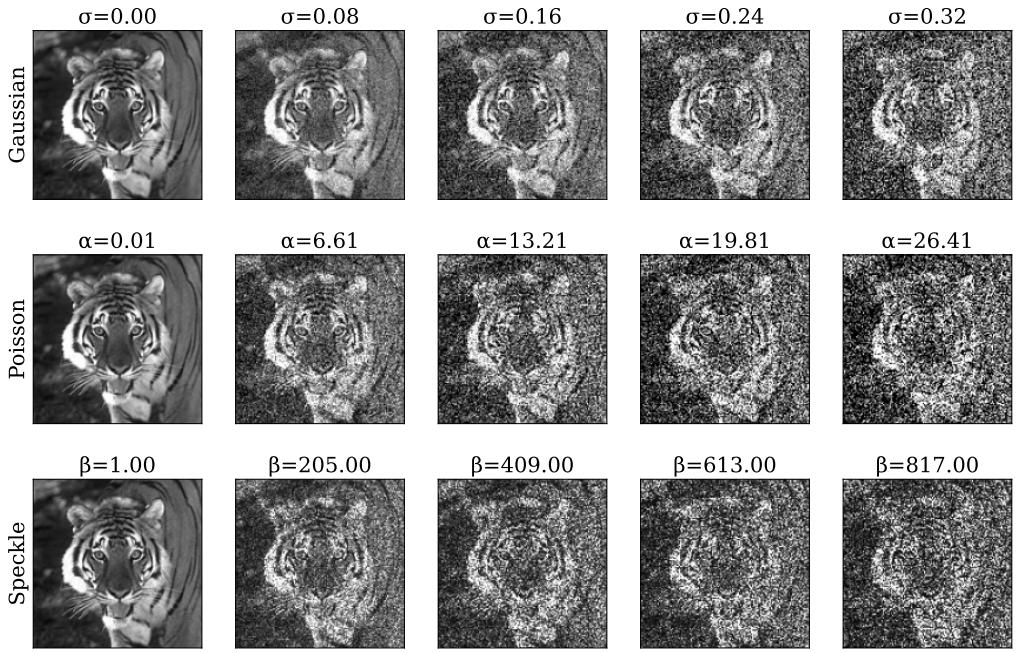

Fig. 1. **Varying the noise specifications.** The first row shows images corrupted by gaussian noise, the second row shows images corrupted by poisson noise, and the last row shows images corrupted by speckle noise. In each of the rows, the other noise parameters are held fixed at 0, 0.01, and 1.00, respectively.

Θ, compared to the ideal function at each point $\theta \in \Theta$, $f^{\theta}_{\text{ideal}}$. More precisely, we want to minimize the maximum gap in performance between f^* and $f^{\theta}_{\text{ideal}}$ across all of Θ. E.g., we want a universal denoiser f^* that works *almost* as well as specialized denoisers $f^{\theta}_{\text{ideal}}$, at all noise levels Θ. Following [5], we can frame this objective as the following optimization problem

$$f^* = \underset{f \in \mathcal{F}}{\text{argmin}} \sup_{\theta \in \Theta} \left\{ \mathcal{L}_f(\theta) - \mathcal{L}_{\text{ideal}}(\theta) \right\}, \tag{3}$$

which we call the *uniform gap problem*.

IV. Proposed Method

A. Adaptive Training

To solve the optimization problem given in (3), [5] proposes rewriting it in its Lagrangian dual formulation and then applying dual ascent, which yields the following iterations:

$$f^{t+1} = \underset{f \in \mathcal{F}}{\text{argmin}} \left\{ \int_{\theta \in \Theta} \mathcal{L}_f(\theta) \lambda^t(\theta) d\theta \right\} \tag{4}$$

$$\lambda^{t+\frac{1}{2}}(\theta) = \lambda^t(\theta) + \gamma^t \left(\frac{\mathcal{L}_{f^{t+1}}}{\mathcal{L}_{\text{ideal}}} - 1 \right) \tag{5}$$

$$\lambda^{t+1}(\theta) = \lambda^{t+\frac{1}{2}}(\theta) / \int_{\theta \in \Theta} \lambda^{t+\frac{1}{2}}(\theta) d\theta, \tag{6}$$

where $\lambda(\theta)$ represents a dual variable at specification $\theta \in \Theta$, and γ is the dual ascent step size. One detail is we use PSNR constraints instead of MSE constraints, which influences the form of Equation 5. Intuitively, the reason is because PSNR is the logarithm of the MSE loss, and a more uniform PSNR gap means that the ratio of the losses is closer to 1, which is where the $\frac{\mathcal{L}_{f^{t+1}}}{\mathcal{L}_{\text{ideal}}} - 1$ term comes from. More details are discussed in the supplement. We can interpret (4) as fitting a model f to the training data, where $\lambda(\theta)$ is the probability of sampling a task at specification θ to draw training data from. Next, (5) updates the sampling distribution $\lambda(\theta)$ based on the difference between the current model f^{t+1}'s performance across $\theta \in \Theta$ and the ideal models' performances. Lastly (6) ensures that $\lambda(\theta)$ is a properly normalized probability distribution. We provide the derivation of the dual ascent iterations from [5] in the supplement.

While Θ has been discussed thus far as a continuum, in practice we sample Θ at discrete locations and compare the model being fit to the ideal model performance at these discrete locations only, so that the cardinality of Θ, $|\Theta|$, is finite. Computing $\mathcal{L}_{\text{ideal}}(\theta)$ for each $\theta \in \Theta$ is extremely computationally demanding if $|\Theta|$ is large; if f is a neural network, it becomes necessary to train $|\Theta|$ neural networks. Furthermore, while $\mathcal{L}_{\text{ideal}}(\theta)$ can be computed offline independent of the dual ascent iterations, during the dual ascent iterations, each update of λ requires the evaluation of $\mathcal{L}_{f^{t+1}}$ for each $\theta \in \Theta$, which is also time intensive if $|\Theta|$ is large.

The key insight underlying our work is that one can approximate $\mathcal{L}_{\text{ideal}}$ and $\mathcal{L}_f$ in order to drastically accelerate the training process.

B. Specification-Loss Landscape Approximations

Let $\mathcal{Q}$ be a class of functions (e.g., quadratics) which we will use to approximate the specification-loss landscape. Instead of computing $\mathcal{L}_{\text{ideal}}(\theta)$ for each $\theta \in \Theta$, we propose instead computing $\mathcal{L}_{\text{ideal}}(\theta)$ at a set of locations $\theta \in \Theta_{\text{sparse}}$, where $|\Theta_{\text{sparse}}| \ll |\Theta|$, and then using these values to form an approximation $\tilde{\mathcal{L}}_{\text{ideal}}$ of $\mathcal{L}_{\text{ideal}}(\theta)$, as

$$\tilde{\mathcal{L}}_{\text{ideal}} = \underset{\tilde{\mathcal{L}} \in \mathcal{Q}}{\text{argmin}} \sum_{\theta \in \Theta_{\text{sparse}}} ||\tilde{\mathcal{L}}(\theta) - \mathcal{L}_{f_{\text{ideal}}}(\theta)||_2^2. \tag{7}$$

We can similarly approximate $\mathcal{L}_{f^{t+1}}$ with a polynomial $\tilde{\mathcal{L}}_{f^{t+1}}$. Then we can solve (3) using dual ascent as before, replacing $\mathcal{L}_{\text{ideal}}$ and $\mathcal{L}_{f^{t+1}}$ with $\tilde{\mathcal{L}}_{\text{ideal}}$ and $\tilde{\mathcal{L}}_{f^{t+1}}$ where appropriate, resulting in a modification to (5):

$$\lambda^{t+\frac{1}{2}} = \lambda^t + \gamma^t \left(\frac{\tilde{\mathcal{L}}_{f^{t+1}}}{\tilde{\mathcal{L}}_{\text{ideal}}} - 1 \right). \tag{8}$$

To justify our use of this approximation, we show that the specification loss-landscapes of the linear subspace projection "denoiser" and the soft-thresholding denoiser are linear and polynomial with respect to their specifications, respectively, and are thus both easy to approximate using polynomials.

First we define our noise model. Let $y = \alpha \text{Poisson}(\frac{1}{\alpha}x_o) + n$ with $n \sim N(0, \sigma^2\mathbf{I})$. First note that if $\frac{1}{\alpha}x_o$ is large the distribution of $\alpha\text{Poisson}(x/\alpha)$ can be approximated with $N(x, \alpha\text{diag}(x))$. Accordingly, $y \approx x + \nu$ where $\nu \sim N(0, \sigma^2\mathbf{I} + \alpha\text{diag}(x))$.

Example 1: Let C denote a k-dimensional subspace of $\mathbb{R}^n$ ($k < n$), and the denoiser be the projection of y onto subspace C denoted by $P_C(y) = \mathbf{P}y$. Then, assuming $\frac{1}{\alpha}x_o$ is large, for every $x_o \in C$

$$\mathbb{E}\|P_C(y) - x_o\|_2^2 \approx k\sigma^2 + \alpha tr(\mathbf{P}diag(x_o)\mathbf{P}^t),$$

where $tr(\cdot)$ denotes the trace. The loss landscape, $\mathcal{L}(\sigma^2, \alpha) = \mathbb{E}\|P_C(y) - x_o\|_2^2$, is linear with respect to σ^2 and α.

Proof: Since the projection onto a subspace is a linear operator and since $P_C(x_o) = x_o$ we have

$$\mathbb{E}\|P_C(y) - x_o\|_2^2 \approx \mathbb{E}\|x_o + P_C(\nu) - x_o\|_2^2 = \mathbb{E}\|P_C(\nu)\|_2^2.$$

Let $r = \mathbf{P}\nu$. Note that $r \sim N(0, \Sigma)$ with $\Sigma = \sigma^2\mathbf{P}\mathbf{P}^t + \alpha\mathbf{P}\text{diag}(x_o)\mathbf{P}^t$. Accordingly,

$$\begin{aligned}\mathbb{E}\|P_C(\nu)\|_2^2 &= \mathbb{E}\|r\|^2 = tr(\Sigma),\\ &= \sigma^2 tr(\mathbf{P}\mathbf{P}^t) + \alpha tr(\mathbf{P}\text{diag}(x_o)\mathbf{P}^t),\\ &= k\sigma^2 + \alpha tr(\mathbf{P}\text{diag}(x_o)\mathbf{P}^t),\end{aligned}$$

where the last equality follows from the fact that $\mathbf{P}\mathbf{P}^t = \mathbf{P}$ and the trace of a k-dimensional projection matrix is k. □

Example 2: Let Γ_k denote the set of k-sparse vectors, $k \ll n$. Let the family of soft-thresholding denoisers be defined as $\eta_\tau(y) = (|y| - \tau)_+ sign(y)$, for $\tau > 0$. Assume that the ground truth signal is bounded, or $||x_o||_\infty \leq c$, for some $c \in \mathbb{R}^+$. Then, assuming $\frac{1}{\alpha}x_o$ is large, for every $x_o \in \Gamma_k$

$$\begin{aligned}\mathbb{E}\|\eta_\tau(y) - x_o\|_2^2 &\leq k(\sigma^2 + c\alpha + \tau^2)\\ &\quad + 2(n-k)\left((\sigma^2+\tau^2)\Phi\left(-\frac{\tau}{\sigma}\right) - \tau\sigma\phi\left(\frac{-\tau}{\sigma}\right)\right),\end{aligned}$$

where $\phi(\cdot)$ denotes the Gaussian density function and $\Phi(\cdot)$ denotes the Gaussian distribution function. The loss landscape, $\mathcal{L}(\sigma^2, \alpha) = \mathbb{E}\|\eta_\tau(y) - x_o\|_2^2$, can be upper bounded closely by a polynomial with respect to σ^2 and α, because ϕ and Φ are smooth functions with convergent Taylor series.

Proof: We have

$$\begin{aligned}\mathbb{E}||\eta_\tau(x_o + \nu) - x_o||_2^2 &= \sum_{i=1}^{k} \mathbb{E}(\eta_\tau(x_{o,i} + \nu_i) - x_{o,i})^2\\ &\quad + (n-k)\mathbb{E}(\eta(\nu_n; \tau))^2\end{aligned}$$

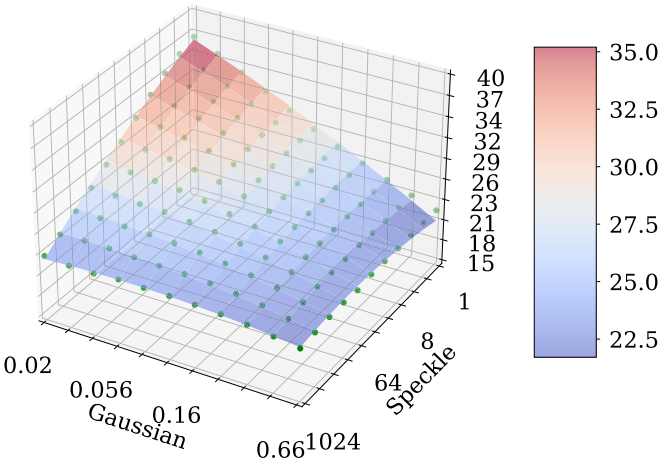

Fig. 2. **Loss landscape visualizations.** PSNR, which we use as our metric for error, versus denoising task specifications. The specification-loss landscapes (which represent the PSNRs a specialized denoiser can achieve at each specification) are smooth and amenable to approximation.

$\mathbb{E}(\eta_\tau(x_{o,i} + \nu_i) - x_{o,i})^2$ is an increasing function of $x_{o,i}$, so we can bound its value from above by taking the limit as

$$\begin{aligned}\mathbb{E}(\eta_\tau(x_{o,i} + \nu_i) - x_{o,i})^2 &\leq \lim_{x_{o,i}\to\infty} \mathbb{E}(\eta_\tau(x_{o,i} + \nu_i) - x_{o,i})^2\\ &= \sigma^2 + c\alpha + \tau^2.\end{aligned}$$

Switching the limit and expectation is justified by the dominated convergence theorem. To calculate the second term, first note that since the ground truth is 0 here there is only Gaussian noise and no Poisson noise. Then, with $\beta = \frac{-\tau}{\sigma}$, we have

$$\begin{aligned}\mathbb{E}(\eta_\tau(\nu_n))^2 &= 2P(\nu_n \leq -\tau)(\mathbb{E}(\nu_i^2 \mid \nu_n \leq -\tau)\\ &\quad + 2\mathbb{E}(\nu_i \mid \nu_n \leq -\tau)\tau + \tau^2)\\ &= 2\Phi(\beta)\left(\sigma^2\left(1 - \beta\frac{\phi(\beta)}{\Phi(\beta)}\right) - 2\sigma\frac{\phi(\beta)}{\Phi(\beta)}\tau + \tau^2\right)\\ &= 2\left((\sigma^2+\tau^2)\Phi\left(-\frac{\tau}{\sigma}\right) - \tau\sigma\phi\left(\frac{-\tau}{\sigma}\right)\right)\end{aligned}$$

where we rewrite the expectations in terms of expressions for the first and second moments of the one-sided truncated Gaussian distribution. Combining the results of these two calculations gives us our desired conclusion. □

The specification-loss landscapes of high-performance image denoisers is highly regular as well. We show the achievable PSNR (peak signal-to-noise ratio) versus noise parameter plots for denoising Gaussian-speckle noise with the DnCNN denoiser [4] in Figure 2. Analogues figures for Poisson-Gaussian and Poisson-speckle noise can be found in the supplement. These landscapes are highly regular and can accurately approximated with quadratic polynomials. (Details on how we validated these quadratic approximations using cross-validation can be found in the supplement.)

Because we're more interested in ensuring a uniform PSNR gap than a more uniform MSE gap, we approximate the ideal PSNRs rather than the ideal mean squared errors. Then, following [5], we convert the ideal PSNRs to mean squared errors with the mapping $\mathcal{L}(\theta) = 10^{-\text{PSNR}(\theta)/10}$ for use in the dual ascent iterations.

C. Exponential Savings in Training-Time

The key distinction between our adaptive training procedure and the adaptive procedure from [5] is that [5] relies upon a dense sampling of the specification-loss landscape whereas our method requires only a sparse sampling of the specification-loss landscape. Because "sampling" points on the specification-loss landscape requires training a CNN, sampling fewer points can result in substantial savings in time and cost.

As one increases the number of specifications, n, needed to describe this landscape ($n = 1$ for Gaussian noise, $n = 2$ for Poisson-Gaussian noise, $n = 3$ for Poisson-Gaussian-Speckle noise, ...), the number points needed to densely sample the landscape grows exponentially. Fortunately, the number of samples needed to fit a quadratic to this landscape only grows quadratically with n: The number of possible nonzero coefficients, i.e., unknowns, of a quadratic of n variables is $\binom{n+2}{2} = \frac{(n+1)(n+2)}{2}$ and thus one can uniquely specify this function from $\frac{(n+1)(n+2)}{2} + 1$ non-degenerate samples. Accordingly, our method has the potential to offer training-time savings that are exponential with respect to the problem specification dimensions. In the next section, we demonstrate our method reduces training-time over $50\times$ for $n = 3$.

V. Synthetic Results

In this section we compare the performance of universal denoising algorithms that are trained (1) by uniformly sampling the noise specifications during training ("uniform"); (2) using the adaptive training strategy from [5] ("dense"), which adaptively trains based on a densely sampled estimate of the loss-specification landscape, and (3) using our proposed adaptive training strategy ("sparse"), which adaptively trains based on a sparsely sampled approximation of the loss-specification landscape. We do not compare against the simple noise level sampling rules such as the 80-20 rule mentioned in [5] which oversamples lower noise levels because it is more difficult to define what region of the space of possible noise levels is considered high versus low noise when there are multiple noise types.

A. Setup

1) Implementation Details: We use a 20-layer DnCNN architecture [4] for all our denoisers. Following [6], we remove all biases from the network layers. We train all of our networks for 50 epochs, with 3000 mini-batches per epoch and 128 image patches per batch, for a total of 384,000 image patches total. We use the Adam optimizer [19] to optimize the weights with a learning rate of 1×10^{-4}, with an L2 loss. During adaptive training, we update the noise level sampling distribution every 10 epochs.

2) Data: To train our denoising models, we curate a high-quality image dataset that combines multiple high resolution image datasets: the Berkeley Segmentation Dataset [20], the Waterloo Exploration Database [21], the DIV2K dataset [22], and the Flick2K dataset [23], which is the same as the dataset used in [24], but different from the one used in [5]. The choice of dataset is not critical to our method, so we use the dataset recommended by the later work, [24]. To test our denoising models, we use the validation dataset from the DIV2K dataset. We use a patch size of 40 pixels by 40 pixels, and patches are randomly cropped from the training images with flipping and rotation augmentations, to generate a total of 384000 patches. All images are grayscale and scaled to the range $[0, 1]$. We use the BSD68 dataset [25] as our testing dataset.

3) Noise Parameters: We consider four types of mixed noise distributions: Poisson-Gaussian, Speckle-Poisson, Speckle-Gaussian, and Speckle-Poisson-Gaussian noise. In each mixed noise type, $\sigma \in [0.02, 0.66]$, $\alpha \in [0.1, 41]$, and $\beta \in [1, 1024]$, and we discretize each range into 10 bins. Note that $B = 1024$ for speckle noise, following the parameterization in Section III-A. These ranges were chosen as they correspond to input PSNRs of roughly 5 to 30 dB. When training the networks with uniform sampling, a new noise specification is first sampled, and then an instance of the noise is sampled from that distribution. The number of training instances is the same as for sparse training, but the proportion of noise distributions seen is different (given by the noise distribution parameter sampling distribution).

4) Sampling the Specification-Loss Landscape: Both our adaptive universal denoiser training strategy ("sparse") and Gnanasambandam and Chan's adaptive universal universal denoiser training strategy ("dense") require sampling the loss-specification landscapes, $\mathcal{L}_{\text{ideal}}(\theta)$, before training. That is, each requires training a collection of denoisers specialized for specific noise parameters $\theta \in \Theta$.

For the loss-specifications landscapes with two dimensions (Poisson-Gaussian, Speckle-Poisson, and Speckle-Gaussian noise), we densely sampling the landscape (i.e., train networks) on a 10×10 grid for the "dense" training. For the "sparse" training (our method) we sample the landscape at 10 random specifications as well as at the specification support's endpoints, for a total of 14 samples for Poisson-Gaussian, Speckle-Poisson, and Speckle-Gaussian noise.

For the loss-specifications landscape with three dimensions (Speckle-Poisson-Gaussian), sampling densely would take nearly a year of GPU hours, so we restrict ourselves to sparsely sampling at only 18 specifications: 10 random locations plus the 8 corners of the specification cube.

5) Training Setup: For each Poisson-Gaussian, Speckle-Poisson, and Speckle-Gaussian noises separately, we (1) use approximations $\tilde{\mathcal{L}}_{\text{ideal}}$ and $\tilde{\mathcal{L}}_f$ to adaptively train a denoiser f^*_{sparse}, (2) use densely sampled landscapes $\mathcal{L}_{\text{ideal}}$ and $\mathcal{L}_f$ to adaptively train a denoiser f^*_{dense}, and (3) sample specifications uniformly to train a denoiser f^*_{uniform}. For Speckle-Poisson-Gaussian noise, the set of possible specifications is too large to train an ideal denoiser for each specification (i.e., compute $\mathcal{L}_{\text{ideal}}$) and so we only provide results for f^*_{sparse} and f^*_{uniform}.

B. Quantitative Results

Tables I, II, III, and IV compare the performance of networks trained with our adaptive training method, which uses sparse samples to approximate of the specification-loss landscape; "ideal" non-blind baselines, which are trained for specific noise parameters; networks trained using the adaptive

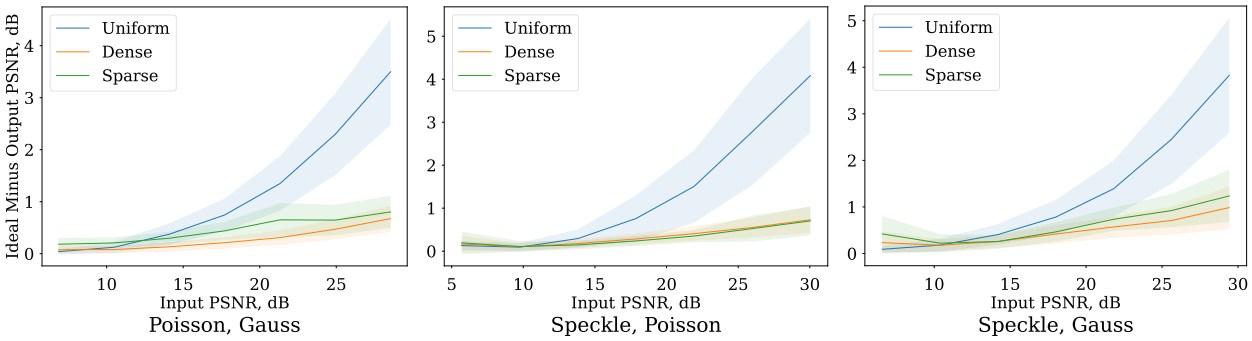

Fig. 3. **Adaptive vs uniform training, 2D specification space.** Adaptive training with sparse sampling and the polynomial approximation works effectively in the 2D problem space and produces a network whose performance is consistently close to the ideal. By contrast, a network trained by uniformly sampling from the space performs far worse than the specialized networks in certain contexts. The error bars represent one standard-deviation. Lower is better.

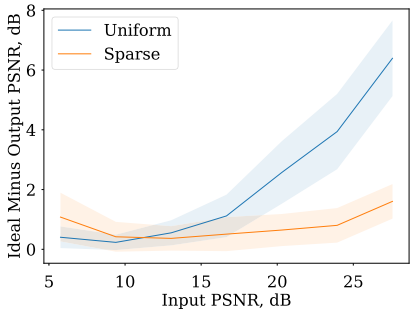

Fig. 4. **Adaptive vs uniform training, 3D specification space.** Adaptive sampling with the polynomial approximation works effectively in the 3D problem space and produces a network whose performance is consistently close to the ideal. By contrast, a network trained by uniformly sampling from the space performs far worse than the specialized networks in certain contexts. The error bars represent one standard-deviation. Lower is better.

TABLE I

QUANTITATIVE COMPARISON OF METHODS ON POISSON-GAUSSIAN NOISE SAMPLED AT VARIOUS LEVELS, USING PSNR (DB)

Poisson	Gaussian	Ideal	Uniform	Dense	Sparse
0.1	0.02	35.3	31.4	34.5	34.6
0.1	0.66	22.0	22.0	21.9	21.9
41	0.02	22.9	22.9	22.9	22.9
41	0.66	21.4	21.3	21.3	21.3
2.0	0.11	27.1	26.6	27.0	27.0

TABLE II

QUANTITATIVE COMPARISON OF METHODS ON SPECKLE-GAUSSIAN NOISE SAMPLED AT VARIOUS LEVELS, USING PSNR (DB)

Speckle	Gaussian	Ideal	Uniform	Dense	Sparse
1.0	0.02	36.6	32.0	35.4	35.1
1.0	0.66	22.0	21.9	21.7	21.6
1024	0.02	23.1	23.1	22.6	22.4
1024	0.66	21.5	21.4	20.8	20.7
32	0.11	27.4	27.0	27.2	27.2

training procedure from [5], which requires training specialized ideal baselines at all noise specifications (densely) beforehand; and networks trained by uniformly sampling the specification-space. Both adaptive strategies approach the performance of the specialized networks and dramatically outperform the uniformly trained networks at certain problem specifications.

TABLE III

QUANTITATIVE COMPARISON OF METHODS ON SPECKLE-POISSON NOISE SAMPLED AT VARIOUS LEVELS, USING PSNR (DB)

Speckle	Poisson	Ideal	Uniform	Dense	Sparse
1.0	0.1	36.1	31.5	35.3	35.3
1.0	41	23.0	22.9	22.9	22.9
1024	0.1	23.0	23.1	23.0	22.9
1024	41	22.0	21.8	21.6	21.6
32	2.0	27.8	27.3	27.6	27.7

TABLE IV

QUANTITATIVE COMPARISON OF METHODS ON SPECKLE-POISSON-GAUSSIAN NOISE SAMPLED AT VARIOUS LEVELS, USING PSNR (DB)

Speckle	Poisson	Gaussian	Ideal	Uniform	Sparse
1	0.1	0.02	34.7	29.3	33.4
1	0.1	0.66	22.0	21.9	21.5
1	41	0.02	23.0	22.9	22.7
1	41	0.66	21.4	21.3	20.8
1024	0.1	0.02	23.2	23.2	22.6
1024	0.1	0.66	21.5	21.4	20.7
1024	41	0.02	22.0	21.9	21.2
1024	41	0.66	21.0	20.8	20.0
64	2.3	0.54	25.8	25.5	25.5

These results are further illustrated in Figures 3 and 4, which report how much the various universal denoisers underperform specialized denoisers across various noise parameters. As hoped, both adaptive training strategies ("dense" and "sparse") produce denoisers which consistently perform nearly as well as the specialized denoisers.[1] In contrast, the uniformly trained denoisers underperform the specialized denoisers in some contexts. Additional quantitative results can be found in the Supplement.

C. Qualitative Results

Figure 5 illustrates the denoisers trained with the different strategies (ideal, uniform, adaptive) on an example corrupted with a low amount of noise and an example corrupted with a high amount of noise. Notice that in the low-noise regime the uniform trained denoiser oversmooths the image so achieves worse performance than the adaptive trained denoiser, whereas

[1]To form these figures we train specialized networks at an additional 100 specifications. These networks were not used for training the universal denoisers.

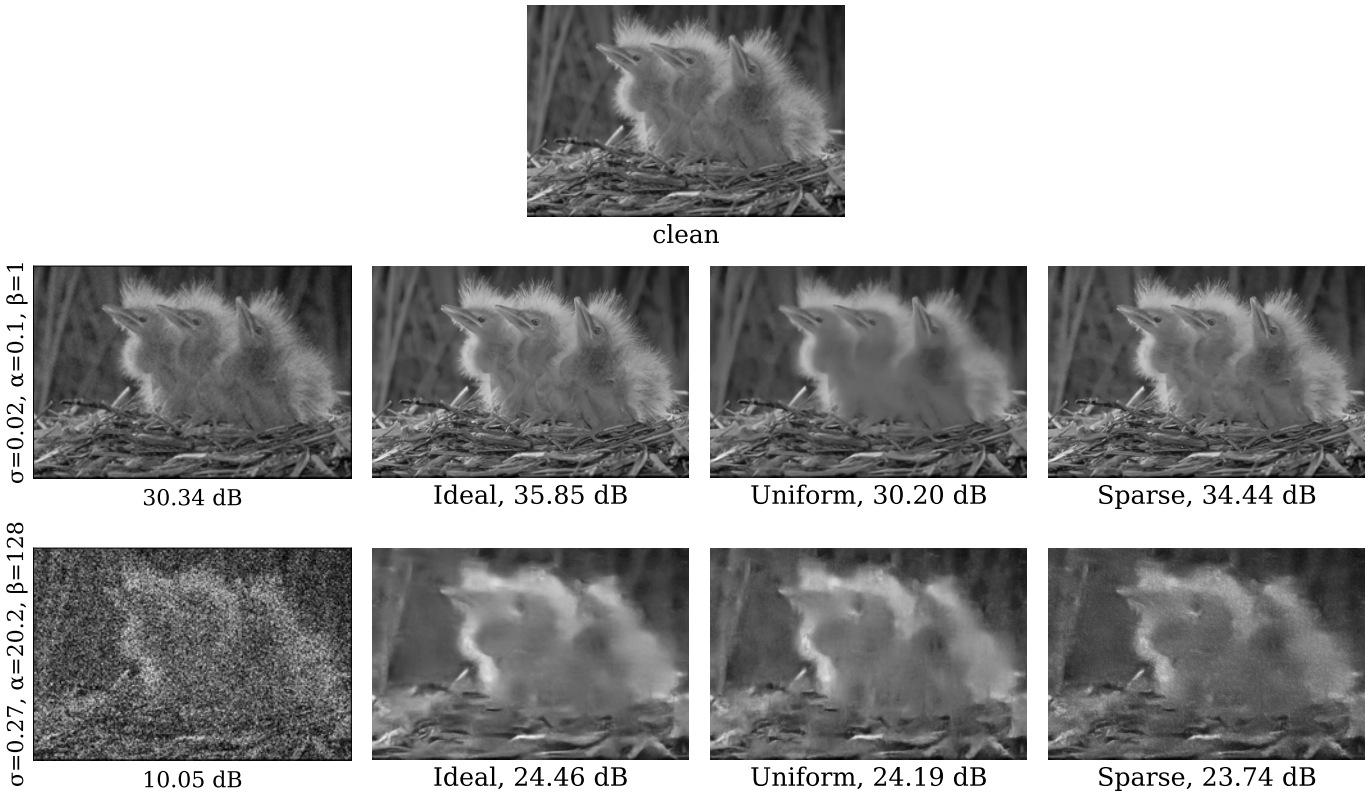

Fig. 5. **Qualitative comparisons, simulated data.** Comparison between the performance of the ideal, uniform-trained, and adaptive distribution, sparse sampling-trained denoisers on a sample image corrupted with a low amount of noise and corrupted with a high amount of noise. Our adaptive distribution sparse approximation based blind training strategy performs only marginally worse than an ideal, non-blind baseline when applied to "easy" problem specifications, and significantly better than the uniform baseline, while also being only marginally worse than an ideal baseline and uniform baseline under "hard" problem specifications.

TABLE V

COMPARING THE OVERALL TRAINING TIMES (IN GPU HOURS) BETWEEN THE VARIOUS UNIVERSAL DENOISER TRAINING STRATEGIES. SPARSELY SAMPLING AND APPROXIMATING THE SPECIFICATION-LOSS RESULTS IN ORDERS OF MAGNITUDE SAVINGS IN TRAINING TIME AND COST

	Uniform	Dense	Sparse
Poisson-Gauss	6.5	891.6	95.2
Speckle-Gauss	7.8	747.8	78.8
Speckle-Poisson	8.0	655.9	70.8
Speckle-Poisson-Gauss	7.5	>7000 (predicted)	138.6

in the high noise regime the uniform trained denoiser outperforms the adaptive trained denoiser. Additional qualitative results can be found in the supplement.

D. Time and Cost Savings

Recall our adaptive training strategy requires pretraining significantly fewer specialized denoisers than the method developed in [5]: 14 vs 100 networks with 2D noise specifications and 18 vs 1000 networks with 3D specifications. We trained the DnCNN denoisers on Nvidia GTX 1080Ti GPUs, which took 6–8 hours to train each network. Overall training times associated with each of the universal denoising algorithms are presented in Table V. In the 3D case, our technique saves over 9 months in GPU compute hours.

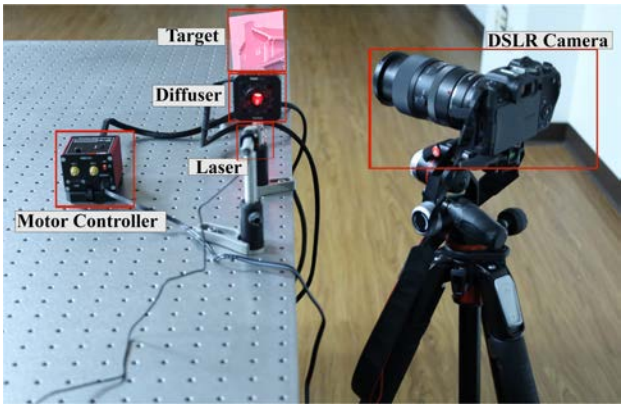

Fig. 6. **Optical system.** We capture images of paper with images on them, illuminated by a laser, using a DSLR camera. The laser beam passes through a diffuser which results in a speckle noise pattern on the image. We rotate the diffuser with a rotation mount connected to a motor controller to gather different independent realizations of speckle.

VI. EXPERIMENTAL VALIDATION

To validate the performance of our method, we experimentally captured a small dataset consisting of 5 scenes with varying amounts of Poisson, Gaussian, and speckled-speckle noise.

A. Optical Setup

Our optical setup consists of a bench-top setup and a DSLR camera mounted on a tripod, pictured in 6. A red laser is

TABLE VI

TWEAKING THE LOSS LANDSCAPE: WE TWEAK THE LOSS LANDSCAPES AS IN THE SETTINGS DESCRIBED IN THE R1QA RESPONSE AND COMPARE OUR ADAPTIVE METHOD WITH TWEAKED LOSS LANDSCAPES TO UNIFORM TRAINING. THE NUMBERS REPORTED ARE THE DIFFERENCE IN PSNR BETWEEN THE IDEAL PSNR AND THE ACHIEVED PSNR; LOWER IS BETTER. OUR METHOD PERFORMERS BETTER INSIDE THE REGIONS WHERE WE BIASED THE SAMPLING, BUT WORSE OUTSIDE THE REGIONS, COMPARED TO UNIFORM SAMPLING

Setting	Uniform		Adaptive	
	Inside	Outside	Inside	Outside
Correlated	3.23	0.22	2.97	0.30
Low	3.13	0.26	3.04	0.50
High	3.41	0.16	3.25	0.43

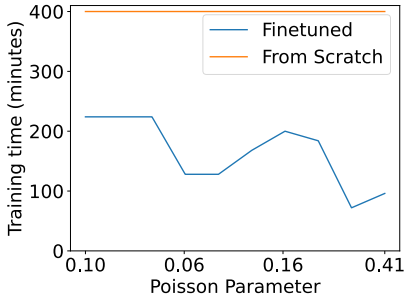

Fig. 7. **Training time of ideal denoisers** Here we compare the time it takes to train a denoiser on one noise configuration from scratch versus finetuned from a general denoiser. In general, we see time savings of 2–4×.

shined through a rotating diffuser to illuminate an image printed on paper. The diffuser is controlled by an external motor controller, which allows us to capture a distinct speckle realization for each image. The resulting signal is captured by the DSLR camera. The pattern on the rotating diffuser causes a speckle noise pattern in the signal. Note, however, that because the paper being imaged is rough with respect to the wavelength of red light, an additional speckle pattern is introduced to the signal—we're capturing speckled-speckle. This optical setup allows us to capture images with varying levels of Gaussian, Poisson, and (speckled) speckle noise. The Gaussian noise can be varied by simply increasing or decreasing the ISO on the camera. Similarly, decreasing exposure time will produce more Poisson noise. Lastly, speckle noise can be varied by changing the number of recorded realizations that are averaged together; as the number of realizations increases, the corruption caused by speckle noise decreases. For each noisy capture, we keep the aperture size fixed at F/11.

B. Data Collection

Using the previously described optical setup, we collected noisy and ground truth images for five different targets. To compute our ground truth image, we capture 16 images with a shutter speed of 0.4 seconds, ISO 100, and F/4 and average them together. To gather the noisy images, for each of the 5 scenes we consider, we capture 16 images at 4 different shutterspeed/ISO pairs, at a fixed aperture of F/11: 1/15 seconds/ISO 160, 1/60 seconds/ISO 640, 1/250 seconds/ISO 2500, and 1/1000 seconds/ISO 10000, for a total of 320 pictures captured. We choose these pairs such that the product of the two settings is approximately the same across settings and thus dynamic range is approximately the same across samples. Additionally, we capture dark frames for each exposure setting and subtract them from the corresponding images gathered. The noisy images were scaled to match the exposure settings of the ground truth image.

C. Performance

1) Quantitative Results: To compare our adaptive sparse training method to the uniform baseline on the real experimental data, we compare the SSIM of the noisy input to the networks with the SSIM of the output, both with respect to the computed ground truth [26]. In this comparison, we use the networks that were trained on the images corrupted with Speckle-Poisson-Gaussian noise. We choose to focus SSIM in this case because it has better perceptual qualities than PSNR, but also include a similar plot using PSNR instead in the supplemental material. We plot the input SSIM minus the output SSIM vs the input SSIM in Figure 10 (lower is better). Even though our adaptive distribution trained networks perform worse than the uniform trained networks in the higher noise regime, the adaptive distribution networks perform better in the low noise regime. Note that for all samples the adaptive trained network improves the image quality, whereas for some lower noise images the uniform trained network actually lowers the image quality. This is because the uniform trained network is oversmoothing the images due to the higher noise images having a large impact on its training loss.

2) Qualitative Results: A qualitative evaluation on real world data collected in our lab suggests that our method effectively extends passed simulation. Close inspection of the samples in 11 show that our method is more consistent at preserving information after denoising at low noise levels. Notably, the uniform denoiser smooths the image, losing high frequency details. Even on highly corrupted samples, the our sparse sampling approach is able to achieve performance close to that of the uniformly trained denoiser. Again, our method retains details that the uniform denoiser smooths out of the image. The drop in performance for our method at high noise levels compared to the uniform sampling strategy is further evidence that the uniform strategy is liable to over-correct for high noise samples.

Additional qualitative results on the experimental data can be found in the supplement.

VII. FURTHER ANALYSIS

We also analyze some possible extensions of our method such as modifying training to address non-uniform distribution of noise specifications, reducing training time by using finetuning, and visualize noise level sampling distributions and PSNR landscapes of networks during training to empirically justify our choice of approximation.

A. Non-Uniform Distribution of Noise Specifications

In practice, some noise specifications/distributions are more likely to occur than others. For instance, [27] demonstrates that

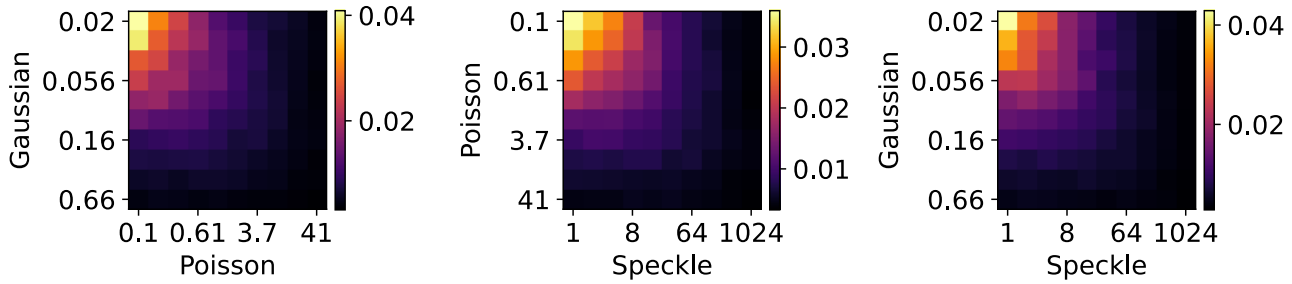

Fig. 8. **Sampling distributions of noise configurations fit by our method** We show the sampling distributions fit by our method for the poisson-gaussian, poisson-speckle, and speckle-gaussian settings. Note that in these sampling distributions, the lower noise configurations have higher probability mass and are preferentially sampled.

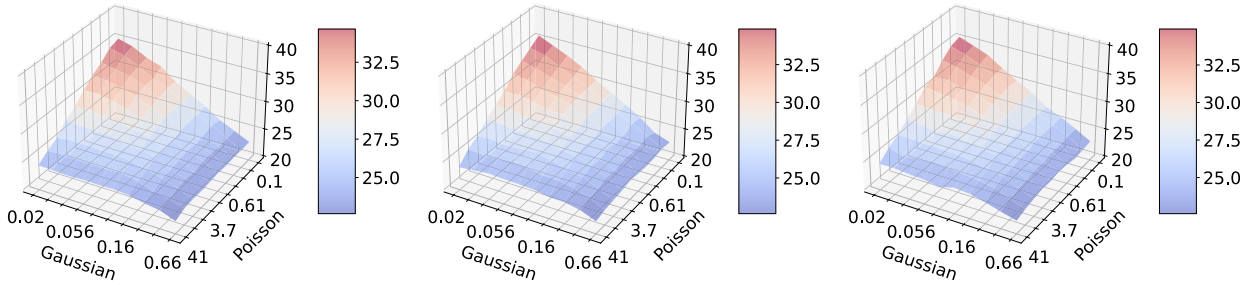

Fig. 9. **PSNR landscape of $\mathcal{L}_{f^{t+1}}$** We show the PSNR landscape of $\mathcal{L}_{f^{t+1}}$ at iterations 10, 30, and 50, three iterations where we update the noise level sampling distribution. They are all smooth.

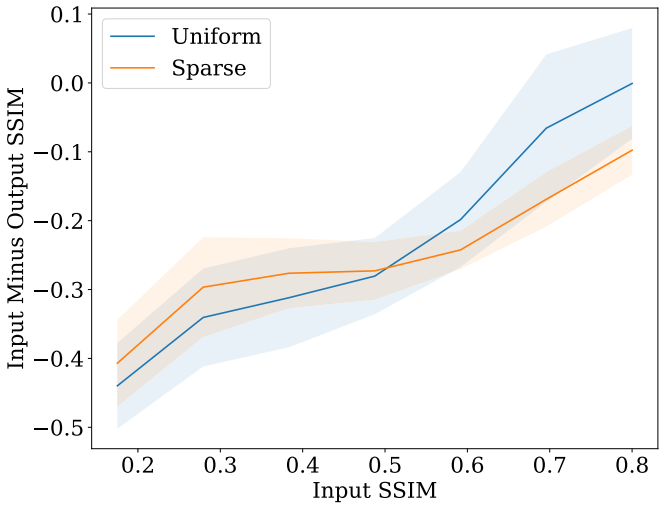

Fig. 10. **Adaptive vs uniform training, experimental data.** Adaptive training with the polynomial approximation from sparse samples works effectively with real data. Our adaptive training strategy consistently outperforms a uniformly trained baseline at lower noise levels while not being significantly worse at higher noise levels (lower input SSIM). Lower is better.

camera auto-exposure settings often operate such that there is a positive correlation between shot and read noise parameters.

If one knows the expected distribution of the noise specifications, i.e., how likely different noise distributions are to be encountered in practice, our universal denoiser strategy can be easily modified to weight the training loss according to each specification's likelihood. Essentially, in order to encourage preferential sampling (and thus improved performance) for certain noise configurations, we can increase the target PSNR in the ideal loss landscape by a constant amount over the configurations we care most about (or are most likely to encounter). This has the effect of prioritizing sampling noise levels from those configurations. To validate this idea, for the case of Poisson-Gaussian noise, we consider three cases where we have prior knowledge about the distribution of noise configurations: 1) Poisson and Gaussian noise are correlated, 2) we are more likely to encounter lower noise levels, and 3) we are more likely to encounter higher noise levels. Specifically, assuming the parameter for Poisson noise is α and for Gaussian noise it is σ, for 1) we add 3 dB to the 10 (α, σ) pairs which are linearly between the extreme points $(0.1, 0.02)$ and $(41, 0.66)$ (preferentially sample correlated noise), for 2) we add 3dB to the 25 pairs where $0.1 \leq \alpha \leq 20.5$ and $0.06 \leq \sigma \leq 0.33$ (preferentially sample weak noise), and for 3) we add 3dB to the 25 pairs where $20.5 \leq \alpha \leq 41$ and $0.33 \leq \sigma \leq 0.66$ (preferentially sample strong noise). The results are reported in Table VI. As expected, our method performs better inside the regions where we biased the sampling, but worse outside the regions, compared to uniform sampling.

B. Finetuning

Fine-tuning is a complementary acceleration strategy which allows one to save a constant factor on the time it takes to calculate $\mathcal{L}_{\text{ideal}}$ by finetuning a general denoiser at each of the noise configurations considered instead of training from scratch. We demonstrate this by considering the 1D version of our problem, where we vary the Poisson noise parameter from 0.1 to 41. The training time for finetuning was determined by stopping when the validation loss matched that of training from scratch for the full amount of time. As can be seen in Figure 7, finetuning saves 2–4× on time compared to training

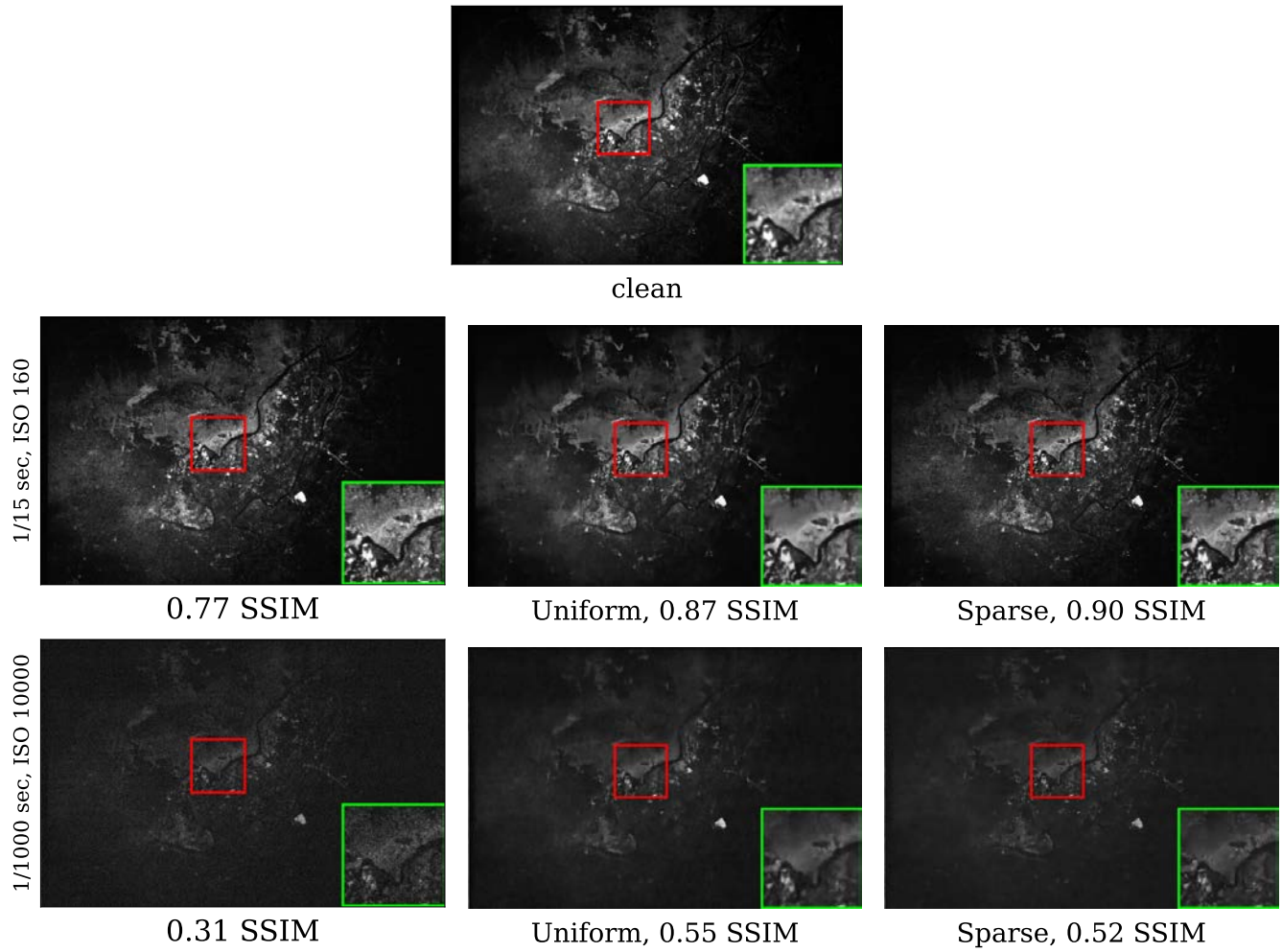

Fig. 11. **Qualitative comparisons, experimental data.** Comparison between the performance of the ideal, uniform-trained, and adaptive distribution, sparse sampling-trained denoisers on a sample image corrupted with a low amount of noise and corrupted with a high amount of noise, as determined by the parameters of our experimental setup. From the closeups it is apparent that the uniform trained denoiser has a tendency to oversmooth its inputs compared to the adaptive distribution trained denoiser. This leads to better performance for the uniform trained at higher noise levels but worse performance at lower noise levels.

from scratch—it is equally applicable to our method and our relative acceleration factor stays the same.

C. Noise Level Sampling Distribution

For the three settings, Speckle-Poisson, Speckle-Gaussian, and Poisson-Gaussian noise, we show the sampling distribution of the noise configurations optimized by our method in Figure 8. Note that our training strategy learns to preferentially sample configurations with lower noise, similar to the 1D case in [5].

D. PSNR Landscapes of Networks During Training

The loss landscapes of $\mathcal{L}_{f^{t+1}}$ are shown in Figure 9 from which it is apparent that the ideal PSNR landscape of $\mathcal{L}_{f^{t+1}}$ is smooth can be well-approximated by a polynomial.

VIII. Conclusion

In this work, we demonstrate that we can leverage a polynomial approximation of the specification-loss landscape to train a denoiser to achieve performance which is uniformly bounded away from the ideal across a variety of problem specifications. Our results extend to high-dimensional problem specifications (e.g., Poisson-Gaussian-Speckle noise) and in this regime our approach offers 50× reductions in training costs compared to alternative adaptive sampling strategies. We experimentally demonstrate our method extends to real-world noise as well. More broadly, the polynomial approximations of the specification-loss landscape developed in this work may provide a useful tool for efficiently training networks to perform a range of imaging and computer visions tasks.

References

[1] G. Ongie, A. Jalal, C. A. Metzler, R. G. Baraniuk, A. G. Dimakis, and R. Willett, "Deep learning techniques for inverse problems in imaging," *IEEE J. Sel. Areas Inf. Theory*, vol. 1, no. 1, pp. 39–56, May 2020.

[2] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[3] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[4] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[5] A. Gnanasambandam and S. Chan, "One size fits all: Can we train one denoiser for all noise levels?" in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 3576–3586.

[6] S. Mohan, Z. Kadkhodaie, E. P. Simoncelli, and C. Fernandez-Granda, "Robust and interpretable blind image denoising via bias-free convolutional neural networks," 2019, *arXiv:1906.05478*.

[7] M. Gharbi, G. Chaurasia, S. Paris, and F. Durand, "Deep joint demosaicking and denoising," *ACM Trans. Graph.*, vol. 35, no. 6, pp. 1–12, Nov. 2016, doi: 10.1145/2980179.2982399.

[8] K. Zhang, W. Zuo, and L. Zhang, "FFDNet: Toward a fast and flexible solution for CNN-based image denoising," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4608–4622, Sep. 2018.

[9] C. A. Metzler and G. Wetzstein, "D-VDAMP: Denoising-based approximate message passing for compressive MRI," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 1410–1414.

[10] A. Q. Wang, A. V. Dalca, and M. R. Sabuncu, "Computing multiple image reconstructions with a single hypernetwork," *Mach. Learn. Biomed. Imag.*, vol. 1, pp. 1–25, Jun. 2022. [Online]. Available: https://melba-journal.org/papers/2022:017.html

[11] W. Tahir, H. Wang, and L. Tian, "Adaptive 3D descattering with a dynamic synthesis network," *Light, Sci. Appl.*, vol. 11, no. 1, Feb. 2022, Art. no. 42, doi: 10.1038/s41377-022-00730-x.

[12] Y.-Q. Wang and J.-M. Morel, "Can a single image denoising neural network handle all levels of Gaussian noise?" *IEEE Signal Process. Lett.*, vol. 21, no. 9, pp. 1150–1153, Sep. 2014.

[13] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, "Plug-and-play priors for model based reconstruction," in *Proc. IEEE Global Conf. Signal Inf. Process.*, Dec. 2013, pp. 945–948.

[14] Y. Romano, M. Elad, and P. Milanfar, "The little engine that could: Regularization by denoising (RED)," *SIAM J. Imag. Sci.*, vol. 10, no. 4, pp. 1804–1844, Jan. 2017, doi: 10.1137/16m1102884.

[15] B. Kawar, G. Vaksman, and M. Elad, "Stochastic image denoising by sampling from the posterior distribution," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 1866–1875.

[16] J. L. Elman, "Learning and development in neural networks: The importance of starting small," *Cognition*, vol. 48, no. 1, pp. 71–99, Jul. 1993, doi: 10.1016/0010-0277(93)90058-4.

[17] R. Gao and K. Grauman, "On-demand learning for deep image restoration," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1095–1104.

[18] J. W. Goodman, *Speckle Phenomena in Optics: Theory and Applications*. Greenwood, CO, USA: Roberts and Company Publishers, 2007.

[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, Y. Bengio and Y. LeCun, Eds., May 2015.

[20] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th IEEE Int. Conf. Comput. Vis.*, vol. 2, Jul. 2001, pp. 416–423.

[21] K. Ma et al., "Waterloo exploration database: New challenges for image quality assessment models," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 1004–1016, Feb. 2017.

[22] E. Agustsson and R. Timofte, "NTIRE 2017 challenge on single image super-resolution: Dataset and study," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1122–1131.

[23] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1132–1140.

[24] K. Zhang, Y. Li, W. Zuo, L. Zhang, L. Van Gool, and R. Timofte, "Plug-and-play image restoration with deep denoiser prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 6360–6376, Oct. 2022.

[25] S. Roth and M. J. Black, "Fields of experts: A framework for learning image priors," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2005, pp. 860–867.

[26] W. Zhou, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, pp. 600–612, 2004. [Online]. Available: http://dblp.uni-trier.de/db/journals/tip/tip13.html#WangBSS04

[27] T. Brooks, B. Mildenhall, T. Xue, J. Chen, D. Sharlet, and J. T. Barron, "Unprocessing images for learned raw denoising," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11028–11037.

Kevin Zhang received the B.A. degree in computer science and pure mathematics from the University of California at Berkeley. He is currently pursuing the Ph.D. degree in computer science with the Intelligent Sensing Laboratory, University of Maryland, College Park, advised by Prof. Christopher Metzler and Jia-Bin Huang. His research focuses on 3D reconstruction from various sensing modalities.

Sakshum Kulshrestha received the B.S. and M.S. degrees in computer science from the University of Maryland. During the master's study, he was based in the Intelligent Sensing Laboratory, where he studied computational imaging and computer vision problems under Dr. Christopher Metzler. Currently, he is with Waymo, where he works on diffusion models and generative approaches to data synthesis.

Christopher A. Metzler (Member, IEEE) is an Assistant Professor with the Department of Computer Science, University of Maryland, College Park, where he leads the UMD Intelligent Sensing Laboratory. He is a member of the University of Maryland Institute for Advanced Computer Studies (UMIACS) and has a courtesy appointment with the Electrical and Computer Engineering Department. His research develops new systems and algorithms for solving problems in computational imaging and sensing, machine learning, and wireless communications. He was an Intelligence Community Postdoctoral Research Fellow, an NSF Graduate Research Fellow, a DoD NDSEG Fellow, and a NASA Texas Space Grant Consortium Fellow. His work has received multiple best paper awards. He recently received the NSF CAREER Award, the AFOSR Young Investigator Program Award, and the ARO Early Career Program Award.