Adaptive Denoising via GainTuning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Deep convolutional neural networks (CNNs) for image denoising are usually trained on large datasets. These models achieve the current state of the art, but they have difficulties generalizing when applied to data that deviate from the training distribution. Recent work has shown that it is possible to train denoisers on a single noisy image. These models adapt to the features of the test image, but their performance is limited by the small amount of information used to train them. Here we propose "GainTuning", in which CNN models pre-trained on large datasets are adaptively and selectively adjusted for individual test images. To avoid overfitting, GainTuning optimizes a single multiplicative scaling parameter (the "Gain") of each channel in the convolutional layers of the CNN. We show that GainTuning improves state-of-the-art CNNs on standard image-denoising benchmarks, boosting their denoising performance on nearly every image in a held-out test set. These adaptive improvements are even more substantial for test images differing systematically from the training data, either in noise level or image type. We illustrate the potential of adaptive denoising in a scientific application, in which a CNN is trained on synthetic data, and tested on real transmission-electron-microscope images. In contrast to the existing methodology, GainTuning is able to faithfully reconstruct the structure of catalytic nanoparticles from these data at extremely low signal-to-noise ratios.

1 Introduction

Like many problems in image processing, the recovery of signals from noisy measurements has been revolutionized by the development of convolutional neural networks (CNNs) [63, 8, 64]. These models are typically trained on large databases of images, either in a supervised [35, 63, 8, 65, 64] or an unsupervised fashion [59, 3, 26, 27]. Once trained, these solutions are evaluated on noisy test images. This approach achieves state-of-the-art performance when the test images and the training data belong to the same distribution. However, when this is not the case, the performance of these models is often substantially degraded [56, 35, 65]. This is an important limitation for many practical applications, in which it is challenging (or even impossible) to gather a training dataset that is comparable in noise and signal content to the images encountered at test time. Overcoming this limitation requires *adaptation* to the test data.

A recent unsupervised method (Self2Self) has shown that CNNs can be trained exclusively on individual test images, producing impressive results [44]. Self2Self adapts to the test image, but its performance is limited by the small amount of information available for learning. As a result it generally underperforms CNN models trained on large databases.

In this work, we propose *GainTuning*, a framework to bridge the gap between models trained on large datasets without adaptation, and models trained exclusively on test images. In the spirit of two recent methods [52, 56], GainTuning adapts pre-trained CNN models to individual test images by minimizing an unsupervised denoising cost function, thus fusing the generic capabilities obtained

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

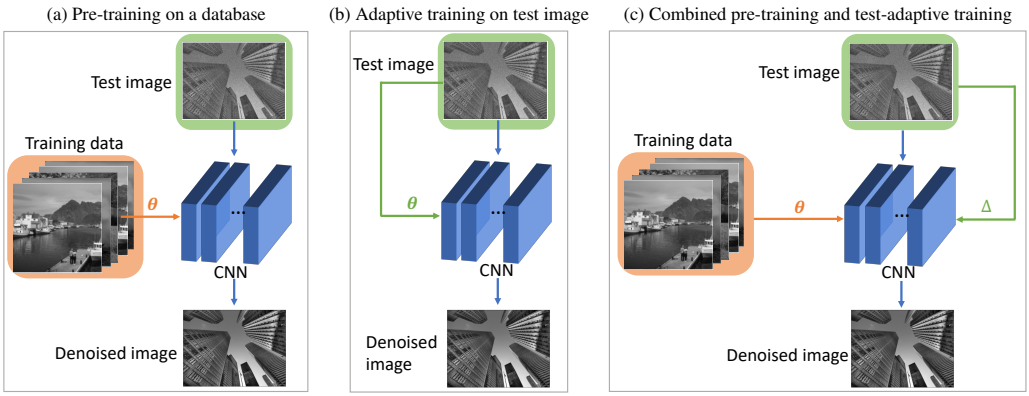


Figure 1: **Proposed denoising paradigm**. (a) Typically, CNNs are trained on a large dataset and evaluated directly on a test image. (b) Recent unsupervised methods perform training on a single test image. (c) We propose GainTuning, a framework which bridges the gap between both of these paradigms: a CNN pre-trained on a large training database is adapted to the test image.

from the training data with specific refinements matched to the structure of the test data. GainTuning does not adapt the full parameter set (filter weights and additive constants) to the test image, but instead optimizes a single multiplicative scaling parameter (the "Gain") for each channel within each layer of the CNN. The dimensionality of this reduced parameter set is a small fraction ($\approx 0.1\%$) of that of the full parameter set. We demonstrate through extensive examples that this prevents overfitting to the test data. The GainTuning procedure is general, and can be applied to any CNN denoising model, regardless of the architecture or pre-training process.

Our contributions. GainTuning provides a novel method for adapting CNN denoisers trained on large datasets to a single test image. GainTuning improves state-of-the-art CNNs on standard image-denoising benchmarks, boosting their denoising performance on nearly every image in held-out test sets. Performance improvements are even more substantial when the test images differ systematically from the training data. We showcase this ability through controlled experiments in which we vary the distribution of the noise and image structure of the test data. Finally, we evaluate GainTuning in a real scientific-imaging application where adaptivity is crucial: denoising transmission-electron-microscope data at extremely low signal-to-noise ratios. As shown in Figure 2, both CNNs pre-trained on simulated images and CNNs trained only on the test data produce denoised images with substantial artefacts. In contrast, GainTuning achieves effective denoising, accurately revealing the atomic structure in the real data.

2 Related Work

Denoising via deep learning. In the last five years, CNN-based methods have clearly outperformed prior state of the art [13, 50, 6, 43, 14, 21, 10]. Denoising CNNs are typically trained in a supervised fashion, minimizing mean squared error between (MSE) over a large database of example ground-truth images and their noisy counterparts [63, 35, 8]. Unsupervised denoising is an alternative approach, which does not rely on ground-truth clean images. There are two main strategies to achieve this: blind-spot methods [26, 27, 3, 59] and techniques based on Stein's unbiased risk estimator (SURE) [13, 31, 46, 34, 52, 53] (see Section 4 for a more detailed description).

Generalization to out-of-distribution images. In order to adapt CNNs to operate on test data with characteristics differing from the training set, recent publications propose fine-tuning the networks using an additional training dataset that is more aligned with the test data [56, 18]. This is a form of transfer learning, a popular technique in classification problems [12, 61]. However, it is often challenging to obtain relevant additional training data. Here, we show that GainTuning can adapt CNN denoisers using only the test image.

Connection to feature normalization in CNNs. Feature normalization techniques such as batch normalization (BN) [23] are a standard component of deep CNNs. BN consists of two stages: (1) centering and normalizing the features corresponding to each channel, (2) scaling and shifting the

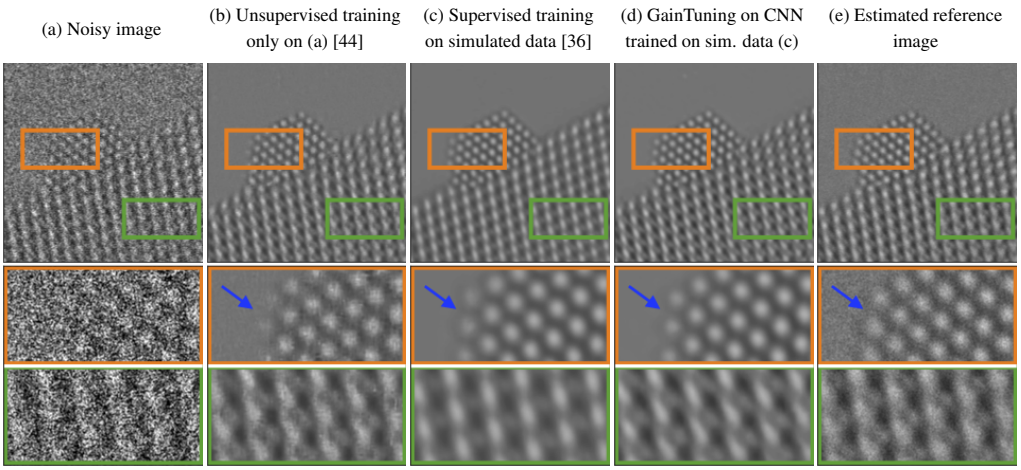


Figure 2: **Denoising results for real-world data.** (a) An experimentally-acquired atomic-resolution transmission electron microscope image of a CeO2-supported Pt nanoparticle. The image has a very low signal to noise ratio (PSNR of $\approx 3dB$). (b) Denoised image obtained using Self2Self [44], which fails to reconstruct three atoms (blue arrow, second row). (c) Denoised image obtained via a CNN trained on a simulated dataset, where the pattern of the supporting atoms is not recovered faithfully (third row). (d) Denoised image obtained by adapting the CNN in (c) to the noisy test image in (a) using GainTuning. Both the nanoparticle and the support are recovered without artefacts. (e) Reference image, estimated by averaging 40 different noisy images of the same nanoparticle.

normalized features using two learned parameters per channel (a scaling factor and a shift). The scaling parameter is analogous to the gain parameter introduced in GainTuning. However, in BN this parameter is fixed during test time, whereas GainTuning explicitly adjusts it based on the test data.

Gain normalization. Motivated by normalization observed in biological sensory neurons [5], adaptive normalization of channel gains has been previously applied in object recognition [24], density estimation [1], and compression [2]. In contrast to these approaches, which adjust gains based on channel responses, GainTuning adjusts them to optimize an unsupervised cost function.

Adapting CNN denoisers to test data. To the best of our knowledge, adaptation of CNN denoisers to test data was pioneered by [53, 56]. Ref. [53] proposes to include the noisy test images in the training set. In a recent extension, the authors propose fine-tuning all the parameters in a pre-trained CNN on a single test image using the SURE cost function [52]. Ref. [56] proposes to do the same using a novel cost function based on noise resampling (see Section 4 for a detailed description). As shown in Section SM 5 fine-tuning all the parameters in a denoiser using only a single test image can lead to overfitting. Ref. [52] avoids this using early stopping, selecting the number of fine-tuning steps beforehand. Ref. [56] uses a specialized architecture with a reduced number of parameters. Here, we propose a framework for adaptive denoising that generalizes these previous methods. We show that several unsupervised cost functions can be used to perform adaptation without overfitting, as long as we only optimize a subset of the model parameters (the gain of each channel).

Adjustment of channel parameters to improve generalization in other tasks. Adjustment of channel parameters, such as gains and biases, has been shown to improve generalization in multiple machine-learning tasks, such as the vision-language problems [41, 11], image generation [7], style transfer [17], and image restoration [20]. In these methods, the adjustment is carried out while training the model by minimizing a supervised cost function. In image classification, recent studies have proposed performing adaptive normalization [25, 39, 48] and optimization [58] of channel parameters during test time, in the same spirit as GainTuning.

3 Proposed Methodology: GainTuning

In this section we describe the GainTuning framework. Let f_θ be a CNN denoiser parameterized by weight and bias parameters, θ. We assume that we have available a training database and a test image

$\mathbf{y}_{\text{test}}$ that we aim to denoise. First, the networks parameters are optimized on the training database

$$\theta_{\text{pre-trained}} = \arg\min_{\theta} \sum_{y \in \text{training database}} \mathcal{L}_{\text{pre-training}}(y, f_\theta(y)). \tag{1}$$

The cost function $\mathcal{L}_{\text{pre-training}}$ used for pre-training can be supervised, if the database contains clean and noisy examples, or unsupervised, if it only contains noisy data.

A natural approach to adapt the pre-trained CNN to the test data is to finetune all the parameters in the CNN, as is done in all prior works on test-time adaptation [56, 52, 18]. Unfortunately this can lead to *overfitting* the test data (see Section SM 5). Due to the large number of degrees of freedom, the model is able to minimize the unsupervised cost function without denoising the noisy test data effectively. This can be avoided to some extent by employing CNN architectures with a small number of parameters [56], or by only optimizing for a short time ("early stopping") [52]. Unfortunately, using a CNN with reduced parameters can limit performance (see Section 5), and it is unclear how to choose a single criterion for early stopping that can operate correctly for each individual test image. Here, we propose a different strategy: tuning a single parameter (the *gain*) in each channel of the CNN. GainTuning can be applied to any pre-trained CNN.

We denote the gain parameter of a channel in a particular layer by $\gamma[\text{layer}, \text{channel}]$, and the parameters of that channel by $\theta_{\text{pre-trained}}[\text{layer}, \text{channel}]$ (this is a vector that contains the weight and bias filter parameters). We define the GainTuning parameters as

$$\theta_{\text{GainTuning}}(\gamma)[\text{layer}, \text{channel}] = \gamma[\text{layer}, \text{channel}]\ \theta_{\text{pre-trained}}[\text{layer}, \text{channel}]. \tag{2}$$

We estimate the gains by minimizing an unsupervised loss that only depends on the noisy image:

$$\hat{\gamma} = \arg\min_{\gamma} \quad \mathcal{L}_{\text{GainTuning}}(\mathbf{y}_{\text{test}}, \theta_{\text{GainTuning}}(\gamma)) \tag{3}$$

The final denoised image is $f_{\theta_{\text{GainTuning}}(\hat{\gamma})}(\mathbf{y}_{\text{test}})$. Section 4 describes several possible choices for the cost function $\mathcal{L}_{\text{GainTuning}}$. Since we use only one scalar parameter per channel, the adjustment performed by GainTuning is very low-dimensional ($\approx 0.1\%$ of the dimensionality of θ). This makes optimization quite efficient, and prevents overfitting (see Section SM 5).

4 Cost Functions for GainTuning

A critical element of GainTuning is the use of an unsupervised cost function, which is minimized in order to adapt the pre-trained CNN to the test data. Here, we describe three different choices, each of which are effective when integrated in the GainTuning framework, but which have different benefits and limitations.

Blind-spot loss. This loss measures the ability of the denoiser to reproduce the noisy observation, while excluding the identity solution. To achieve this, the CNN must estimate the jth pixel y_j of the noisy image y as a function of the other pixels $y_{\{j\}^c}$, but *excluding the pixel itself.* As long as the noise degrades pixels *independently*, this will force the network to learn a nontrivial denoising function that exploits the relationships between pixels that are present in the underlying clean image(s). The resulting loss can be written as

$$\mathcal{L}_{\text{blind-spot}}(\mathbf{y}, \theta) = \mathbb{E}\left[(f_\theta(\mathbf{y}_{\{j\}^c})_j - \mathbf{y}_j)^2\right]. \tag{4}$$

Here the expectation is over the data distribution and the selected pixel. We have slightly abused notation to indicate that the network f_θ does not use y_j. This "blind spot" can be enforced through architecture design [27], or by masking [3, 26] (see also [44] and [59] for related approaches). The blind-spot loss has a key property that makes it very powerful in practical applications: it makes no assumption about the noise distribution (other than pixel-wise independence). When combined with GainTuning it achieves effective denoising of real electron-microscope data at very low SNRs (see Figure 2 and Section 5.4, SM 6.5).

Stein's Unbiased Risk Estimator (SURE). Let $\mathbf{x}$ be an N-dimensional ground-truth random vector $\mathbf{x}$ and let $\mathbf{y} := \mathbf{x} + \mathbf{n}$ be a corresponding noisy observation, where $\mathbf{n} \sim \mathcal{N}(0, \sigma_n^2 \mathbf{I})$. SURE provides an expression for the MSE between $\mathbf{x}$ and a denoised estimate $f_\theta(\mathbf{y})$, which *only depends on the noisy observation* $\mathbf{y}$:

$$\mathbb{E}_{\mathbf{x},\mathbf{y}}\left[\frac{1}{N}\left\|\mathbf{x} - f_\theta(\mathbf{y})\right\|^2\right] = \mathbb{E}_{\mathbf{y}}\left[\frac{1}{N}\left\|\mathbf{y} - f_\theta(\mathbf{y})\right\|^2 - \sigma^2 + \frac{2\sigma^2}{N}\sum_{k=1}^{N}\frac{\partial(f_\theta(\mathbf{y})_k)}{\partial \mathbf{y}_k}\right] := \mathcal{L}_{\text{SURE}}(\mathbf{y}, \theta). \tag{5}$$

The last term in Equation 5 is the divergence of f_θ, which can be approximated using Monte Carlo techniques [45] (Section SM 4). The divergence is the sum of the partial derivatives of each denoised pixel with respect to the corresponding input pixel. Intuitively, penalizing it forces the denoiser to not rely as heavily on the jth noisy pixel to estimate the jth clean pixel. This is similar to the blind-spot strategy, with the added benefit that the jth noisy pixel is not ignored completely. To further illustrate this connection, consider a linear convolutional denoising function $f_\theta(\mathbf{y}) = \theta \circledast \mathbf{y}$, where the center-indexed parameter vector is $\theta = [\theta_{-k}, \theta_{-k+1}, \ldots, \theta_0, \ldots, \theta_{k-1}, \theta_k]$. The SURE cost function (Equation 5) reduces to

$$\mathbb{E}_{\mathbf{n}} \left[\frac{1}{N} \left\| \mathbf{y} - \theta \circledast \mathbf{y} \right\|^2 \right] - \sigma^2 + 2\sigma^2 \theta_0 \tag{6}$$

The SURE loss equals the MSE between the denoised output and the noisy image, with a penalty on the "self" pixel. As this penalty is increased, the self pixel will be ignored, so the loss tends towards the blind-spot cost function. When integrated in the GainTuning framework, the SURE loss achieves effective denoising in the case of additive Gaussian noise, outperforming the blind-spot loss.

Noise Resampling. Ref. [56] introduced a novel procedure for adaptation which we call *noise resampling*. Given a pre-trained denoiser f_θ and a test image $\mathbf{y}$, first we obtain an initial denoised image by applying f_θ to $\mathbf{y}$, $\hat{\mathbf{x}} := f_{\theta_{\text{pre-trained}}}(\mathbf{y})$. Then we artificially corrupt $\hat{\mathbf{x}}$ by simulating noise from the same distribution as the data of interest to create synthetic noisy examples. Finally, the denoiser is fine-tuned by minimizing the MSE between $\hat{\mathbf{x}}$ and the synthetic examples. If we assume additive noise, the resulting loss is of the form

$$\mathcal{L}_{\text{noise resampling}}(\mathbf{y}, \theta) = \mathbb{E}_n \left[\| (f_\theta(\hat{\mathbf{x}} + \mathbf{n}) - \hat{\mathbf{x}} \|^2 \right]. \tag{7}$$

Noise resampling is reminiscent of Refs. [38, 60], which add noise to an already noisy image. When integrated in the GainTuning framework, the noise-resampling loss results in effective denoising in the case of additive Gaussian noise, although it tends to underperform the SURE loss.

5 Experiments and Results

In order to evaluate the performance of GainTuning we performed three different types of experiment: **In-distribution** (test examples held out from the training set); **out-of-distribution noise** (noise level or distribution differs between training and test examples); and **out-of-distribution signal** (clean images drawn from a different set than the training set). We also apply GainTuning to **real data** from a transmission electron microscope. Our experiments make use of four **datasets**: The BSD400 natural image database [32] with test sets Set12 and Set68 [63], the Urban100 images of urban environments [22], the IUPR dataset of scanned documents [4], and a set of synthetic piecewise constant images [29] (see Section SM 2). We demonstrate the broad applicability of GainTuning by using it in conjunction with multiple **architectures for image denoising**: DnCNN [63], BFCNN [35], UNet [47] and Blind-spot net [27] (see Section SM 1). Finally, we compare our results to several **baselines**: (1) models trained on the training database, (3) CNN models adapted by fine-tuning all parameters (as opposed to just the gains), (3) a model trained only on the test data, (4) LIDIA, a specialized architecture and adaptation strategy proposed in [56]. We provide details on training and optimization in Section SM 3.

5.1 GainTuning surpasses state-of-the-art performance for in-distribution data

Experimental set-up. We use BSD400, a standard natural-image benchmark, corrupted with Gaussian white noise with standard deviation σ sampled uniformly from $[0, 55]$ (relative to pixel intensity range $[0, 255]$). Following [63], we evaluate performance on two independent test sets: Set12 and BSD68, corrupted with Gaussian noise with $\sigma \in \{30, 40, 50\}$.

Comparison to pre-trained CNNs. GainTuning consistently improves the performance of pre-trained CNN models. Figure 3 shows this for two different models, DnCNN [63] and UNet [47] (see also Section SM 6.1). The SURE loss outperforms the blind-spot loss, and is slightly better than noise resampling (Table 2). The same holds for other architectures, as reported in Section SM 6.1. On average the improvement is modest, but for some images it is quite substantial (up to 0.3 dB in PSNR for $\sigma = 30$, see histogram in Figure 3).

	Model		Set12			BSD68		
			$\sigma = 30$	40	50	30	40	50
GainTuning	DnCNN	Pre-trained	29.52	28.21	27.19	28.39	27.16	26.27
		GainTuning	**29.62**	**28.30**	**27.29**	**28.47**	**27.23**	**26.33**
	UNet	Pre-trained	29.34	28.05	27.05	28.27	27.05	26.15
		GainTuning	29.46	28.15	27.13	28.34	27.12	26.22
Baseline	LIDIA	Pre-trained	29.46	27.95	26.58	28.24	26.91	25.74
		Adapted	29.50	28.10	26.95	28.23	26.97	26.02
	Self2Self		29.21	27.80	26.58	27.83	26.67	25.73

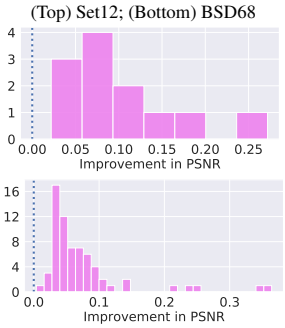


Figure 3: **GainTuning achieves state-of-the-art performance.** (Left) The average PSNR on two test set of generic natural images improves after GainTuning for different architectures across multiple noise levels. The CNNs are trained on generic natural images (BSD400). (Right) Histograms showing improvement in performance for each image in each of the two test sets at $\sigma = 30$.

Comparison to other baselines. GainTuning outperforms fine-tuning based on optimizing all the parameters for different architectures and loss functions (see Section SM 5). GainTuning clearly outperforms a Self2Self model, which is trained exclusively on the test data (Figure 3). It also outperforms the specialized architecture and adaptation process introduced in [56], with a larger gap in performance for higher noise levels.

5.2 GainTuning generalizes to new noise distributions

Experimental set-up. The same set-up as Section 5.1 is used, except that the test sets are corrupted with Gaussian noise with $\sigma \in \{70, 80\}$ (recall that training occurs with $\sigma \in [0, 55]$).

Comparison to pre-trained CNNs. Pre-trained CNN denoisers fail to generalize in this setting. GainTuning consistently improves their performance (see Figure 4 and).

The SURE loss again outperforms the blind-spot loss, and is slightly better than noise resampling (see Section SM 6.2). The same holds for other architectures, as reported in Section SM 6.2. The improvement in performance for all images is substantial (up to 12 dB in PSNR for $\sigma = 80$, see histogram in Figure 4).

Comparison to other baselines. GainTuning achieves comparable performance to a gold-standard CNN trained with supervision at all noise levels (Figure 4). GainTuning matches the performance of a bias-free CNN [35] specifically designed to generalize to out-of-distribution noise (Figure 4). GainTuning outperforms fine-tuning based on optimizing all the parameters for different architectures and loss functions (see Section SM 5). GainTuning clearly outperforms a Self2Self model trained exclusively on the test data (Section SM 6.2), and the LIDIA adaptation method [56].

5.3 GainTuning generalizes to out-of-distribution image content

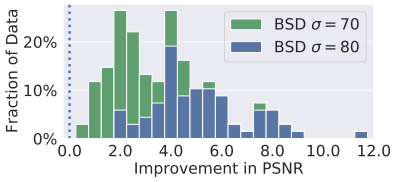
Experimental set-up. We evaluate the performance on of GainTuning on test images that have different characteristics from the training images. We perform the following controlled experiments:

(a) Simulated piecewise constant images $\rightarrow$ Natural images. We pre-train CNN denoisers on simulated piecewise constant images. These images consists of constant regions (of different intensities values) with the boundaries having varied shapes such as circle and lines with different orientations (see Section SM 2 for some examples). Piecewise constant images provide a crude model for natural images [33, 42, 29]. We use GainTuning to adapt a CNN trained on this dataset to generic natural images (Set12). This experiment demonstrates the ability of GainTuning to adapt from a simple simulated dataset to a significantly more complex real dataset.

(b) Generic natural images $\rightarrow$ Images with high self-similarity. We apply GainTuning to adapt a CNN trained on generic natural images to images in Urban100 dataset. Urban100 consists of images of buildings and other structures typically found in an urban setting, which contain substantially more repeating/periodic structure (see Section SM 2) than generic natural images.

Out-of-distribution test noise

Test set	σ	Trained on $\sigma \in [0, 55]$		Bias Free	Trained on
		Pre-trained	Gaintuning	Model [35]	$\sigma \in [0, 100]$
Set12	70	22.45	25.54	25.59	25.50
	80	18.48	24.57	24.94	24.88
BSD68	70	22.15	24.89	24.87	24.88
	80	18.72	24.14	24.38	24.36



Out-of-distribution test image

	Training data	Test data	Pre-trained	Gaintuning
(a)	Piecewise constant	Natural images	27.31	28.60
(b)	Natural images	Urban images	28.35	28.79
(c)	Natural images	Scanned documents	30.02	30.73

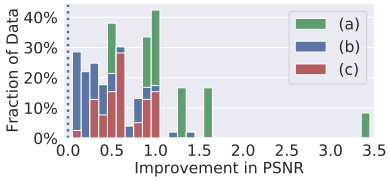


Figure 4: **GainTuning generalizes to out-of-distribution data.** Average performance of a CNN trained to denoise at noise levels $\sigma \in [0, 55]$ improves significantly on test image with noise outside the training range, $\sigma = 70, 80$ (top) and on images with different characteristics than training data (bottom) after GainTuning. Capability of GainTuning to generalize to out-of-distribution noise is comparable to that of Bias-Free CNN [35], which is an architecture explicitly designed to generalize to noise levels outside the training range, and to that of a denoiser trained with supervision at all noise levels. (Right) Histogram showing improvement in performance for each image in the test set. The improvement is substantial across most images, reaching nearly 12dB improvement in one example.

(c) Generic natural images $\rightarrow$ Images of scanned documents. We apply GainTuning to adapt a CNN trained on generic natural images to images of scanned documents in IUPR dataset (see Section SM 2).

All CNNs were trained for denoising Gaussian white noise with standard deviation $\sigma \in [0, 55]$ and evaluated at $\sigma = 30$.

Comparison to pre-trained CNNs. GainTuning consistently improves the performance of pre-trained CNNs in all the three experiments. Figure 4 shows this for DnCNN when GainTuning is based on SURE loss. We obtain similar results with other architectures (see Section SM 6.3). In experiment (a), all test images show substantial improvements, with one image improving as much as 3 dB in PNSR (at $\sigma = 30$). We observe similar trends for experiments (b) and (c) as well, with improvements being better on an average for experiment (c). Note that we obtain similar performance increases when both *image and noise are out-of-distribution* as discussed in Section SM 6.4.

Comparison to other baselines. Experiment (a): (1) GainTuning outperforms optimizing all the parameters over different architectures and loss functions for experiment (Section SM 5). (2) Self2Self trained only on test data outperforms GainTuningin these cases, even though GainTuning improves the performance of pre-trained CNN about 1.3 dB on an average. This is because the test images contain content that differs substantially from the training images. Self2Self provides the strongest form of adaptation, since it is trained exclusively on the test image, whereas the denoising properties of GainTuning are partially due to the pretraining (see Sections 7, SM 6.3). (3) We did not evaluate LIDIA [56] for this experiment. Experiments (b) and (c). (1) Training all parameters clearly outperforms GainTuning for case (b), but has similar performance for (c). GainTuning outperforms LIDIA on experiments (b) and (c). Self2Self trained exclusively on test data outperforms GainTuning(and LIDIA) on (b) and (c) (see Sections 7, SM 6.3).

5.4 Application to Electron microscopy

Scientific motivation. Transmission electron microscopy (TEM) is a popular imaging technique in materials science [51, 55]. Recent advancements in TEM enable to image at high frame rates [16, 15]. These images can for example capture the dynamic, atomic-level rearrangements of catalytic systems [54, 19, 28, 30, 9], which is critical to advance our understanding of functional materials. Acquiring image series at such high temporal resolution produces data severely degraded by shot noise. Consequently, there is an acute need for denoising in this domain.

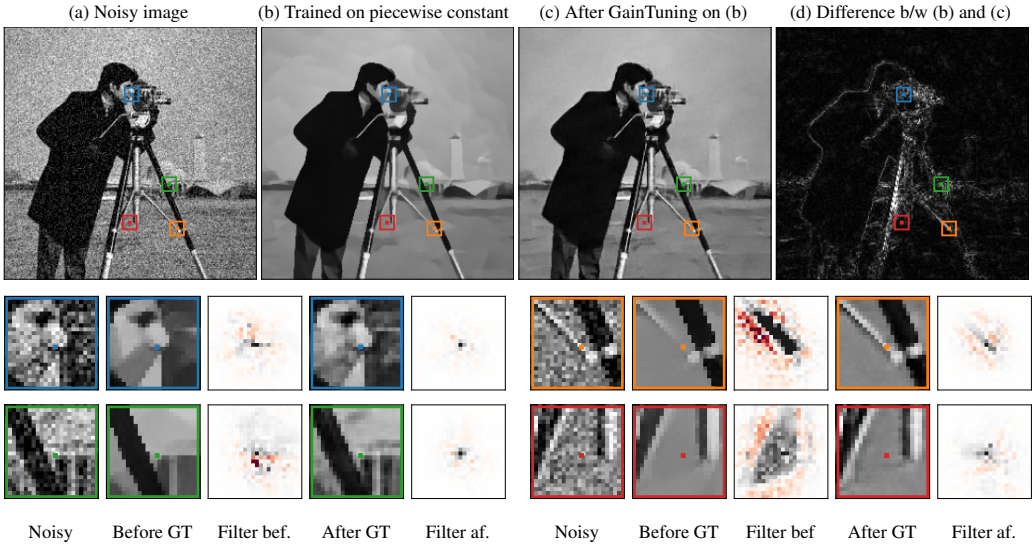


Figure 5: **Adaptation to new image content**. (Top) A CNN pre-trained on piecewise constant images applied to a natural test image (a) oversmooths the image and blurs the details (b), but is able to recover more detail after applying GainTuning (c). (Bottom) The CNN estimates a denoised pixel (dot at the center of each image) as a linear combination of the noisy input pixels. The weighting functions (filters) of pre-trained CNN are more dispersed, consistent with the training set. However, after GainTuning, the weighting functions are more precisely targeted to the local features, resulting in a denoised image (c) with more details.

The need for adaptive denoising. Ground-truth images are not available in TEM, because measuring at high SNR is often impossible. Prior works have addressed this by using simulated training data [36, 57], whereas others have trained CNNs directly on noisy real data [49].

Dataset. We use the training set of 5583 simulated images and the test set of 40 real TEM images from [36, 57]. The data correspond to a catalytic platinum nanoparticle on a CeO_2 support (Section SM 2).

Comparison to pre-trained CNN. A CNN [27] pre-trained on the simulated data fails to reconstruct the pattern of atoms faithfully (green box in Figure 2 (c), (e)). GainTuning applied to this CNN using the blind-spot loss correctly recovers this pattern (green box in Figure 2 (d), (e)) reconstructing the small oxygen atoms in the CeO_2 support. GainTuning with noise resampling failed to reproduce the support pattern (probably because it is absent from the initial denoised estimate) (Section SM 6.5).

Comparison to other baselines. GainTuning clearly outperforms Self2Self, which is trained exclusively on the real data. The denoised image from Self2Self shows missing atoms and substantial artefacts (see Section SM 6.5). We also compare GainTuning dataset to blind-spot methods using the 40 test frames [27, 49]. GainTuning clearly outperforms these methods (see Section SM 6.5). Finally, GainTuning outperforms fine-tuning based on optimizing all the parameters, which overfits heavily (see Section SM 5).

6 Analysis

In this section, we perform a qualitative analysis of the properties of GainTuning.

What kind of images benefit the most from adaptive denoising? Section SM 7.1 shows the images in the different test datasets for which GainTuning achieves the most and the least improvement in PSNR. For different architecture, the result is quite consistent: GainTuning is more beneficial for images with highly repetitive patterns. This makes intuitive sense; the repetitions effectively provide multiple examples from which to learn these patterns during the unsupervised refinement.

Generalization via GainTuning. Section SM 7.2 shows that GainTuning can achieve generalization to images that are similar to the test image used for adaptation.

How does GainTuning adapt to out-of-distribution noise? Generalization to out-of-distribution noise provides a unique opportunity to understand how GainTuning modifies the denoising function. Ref. [35] shows that the first-order Taylor approximation of denoising CNNs trained on multiple noise levels tend to have a negligible constant term, and that the growth of this term is the primary culprit for the failure of these models when tested on new noise levels. GainTuning reduces the amplitude of this constant term, facilitating generalization (See Section SM 7.3 for more details).

How does GainTuning adapt to out-of-distribution images? Figure 5 shows the result of applying a CNN trained on piecewise-constant images to natural images. Due to its learned prior, the CNN averages over large areas, ignoring fine textures. This is apparent in the equivalent linear filters obtained from a local linear approximation of the denoising function [35]. After GainTuning the model learns to preserve the fine features much better, which is reflected in the equivalent filters (see Section SM 7.4 for more details).

7 Limitations

As shown in Section 5, GainTuning improves the state-of-the-art on benchmark datasets, adapts well to out-of-distribution noise and image content, and outperforms all existing methods on an application to real world electron-microscope data. A crucial component in the success of GainTuning is restricting the parameters that are optimized at test time. However, this constraint also limits the potential improvement in performance one can achieve, as seen when fine-tuning for test images from the Urban100 and IUPR datasets, each of which contain many images with highly repetitive structure. In these cases, we observe that fine-tuning all parameters, and even training only on the test data using Self2Self often outperforms GainTuning. This raises the question of how to effectively leverage training datasets for such images.

In addition, when the pre-trained denoiser is highly optimized, and the test image is within distribution, GainTuning can sometimes slightly degrade the performance of the denoiser. This is atypical (3 occurrences in 412 GainTuning experiments using DnCNN and SURE), and the decreases are quite small (maximum PSNR degradation of about 0.02dB, compared to maximum improvement of nearly 12dB).

8 Conclusions

We've introduced GainTuning an adaptive denoising methodology for adaptively fine-tuning a pre-trained CNN denoiser on individual test images. The method, which is general enough to be used with any denoising CNN, improves the performance of state-of-the-art CNNs on standard denoising benchmarks, and provides even more substantial improvements when the test data differ systematically from the training data, either in noise level or image type. We demonstrate the potential of adaptive denoising in scientific imaging through an application to electron microscopy. Here, GainTuning is able to jointly exploit synthetic data and test-time adaptation to reconstruct meaningful structure (the atomic configuration of a nanoparticle and its support), which cannot be recovered through alternative approaches. We hope that these results will motivate further development of techniques for test-time adaptation in denoising and other image-processing tasks. A concrete challenge for future research is how to exploit the unsupervised denoising strategy in Self2Self, which relies heavily on dropout and ensembling, in combination with pre-trained models. Finally, we would like to comment on the potential negative societal outcomes of our work. The training of CNN models on large clusters contributes to carbon emissions, and therefore global warming. In addition, the black-box characteristics of these models may result in unintended biases. We hope that these effects may be offset to some extent by the potential applications of these approaches to tackle challenges such as global warming. In fact, the catalytic system studied in this work is representative of catalysts used in clean energy conversion and environmental remediation [37, 62, 40].

References

[1] BALLÉ, J., LAPARRA, V., AND SIMONCELLI, E. P. Density modeling of images using a generalized normalization transformation. *arXiv preprint arXiv:1511.06281* (2015).

[2] BALLÉ, J., LAPARRA, V., AND SIMONCELLI, E. P. End-to-end optimized image compression. *arXiv preprint arXiv:1611.01704* (2016).

[3] BATSON, J., AND ROYER, L. Noise2self: Blind denoising by self-supervision. In *Proceedings of the 36th International Conference on Machine Learning* (2019), pp. 524–533.

[4] BUKHARI, S. S., SHAFAIT, F., AND BREUEL, T. M. The iupr dataset of camera-captured document images. In *International Workshop on Camera-Based Document Analysis and Recognition* (2011), Springer, pp. 164–171.

[5] CARANDINI, M., AND HEEGER, D. J. Normalization as a canonical neural computation. *Nature Reviews Neuroscience 13*, 1 (2012), 51–62.

[6] CHANG, S. G., YU, B., AND VETTERLI, M. Adaptive wavelet thresholding for image denoising and compression. *IEEE Trans. Image Processing 9*, 9 (2000), 1532–1546.

[7] CHEN, T., LUCIC, M., HOULSBY, N., AND GELLY, S. On self modulation for generative adversarial networks. *arXiv preprint arXiv:1810.01365* (2018).

[8] CHEN, Y., AND POCK, T. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE transactions on pattern analysis and machine intelligence 39*, 6 (2016), 1256–1272.

[9] CROZIER, P. A., LAWRENCE, E. L., VINCENT, J. L., AND LEVIN, B. D. Dynamic restructuring during processing: approaches to higher temporal resolution. *Microscopy and Microanalysis 25*, S2 (2019), 1464–1465.

[10] DABOV, K., FOI, A., KATKOVNIK, V., AND EGIAZARIAN, K. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on Image Processing* (2017), 2080–2095.

[11] DE VRIES, H., STRUB, F., MARY, J., LAROCHELLE, H., PIETQUIN, O., AND COURVILLE, A. Modulating early visual processing by language. *arXiv preprint arXiv:1707.00683* (2017).

[12] DONAHUE, J., JIA, Y., VINYALS, O., HOFFMAN, J., ZHANG, N., TZENG, E., AND DARRELL, T. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning* (2014), PMLR, pp. 647–655.

[13] DONOHO, D., AND JOHNSTONE, I. Adapting to unknown smoothness via wavelet shrinkage. *J American Stat Assoc 90*, 432 (December 1995).

[14] ELAD, M., AND AHARON, M. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. on Image processing 15*, 12 (2006), 3736–3745.

[15] ERCIUS, P., JOHNSON, I., BROWN, H., PELZ, P., HSU, S.-L., DRANEY, B., FONG, E., GOLDSCHMIDT, A., JOSEPH, J., LEE, J., AND ET AL. The 4d camera – a 87 khz frame-rate detector for counted 4d-stem experiments. *Microscopy and Microanalysis* (2020), 1–3.

[16] FARUQI, A., AND MCMULLAN, G. Direct imaging detectors for electron microscopy. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 878* (2018), 180 – 190. Radiation Imaging Techniques and Applications.

[17] GHIASI, G., LEE, H., KUDLUR, M., DUMOULIN, V., AND SHLENS, J. Exploring the structure of a real-time, arbitrary neural artistic stylization network. *arXiv preprint arXiv:1705.06830* (2017).

[18] GONG, K., GUAN, J., LIU, C.-C., AND QI, J. Pet image denoising using a deep neural network through fine tuning. *IEEE Transactions on Radiation and Plasma Medical Sciences 3*, 2 (2018), 153–161.

[19] GUO, H., SAUTET, P., AND ALEXANDROVA, A. N. Reagent-triggered isomerization of fluxional cluster catalyst via dynamic coupling. *The Journal of Physical Chemistry Letters 11*, 8 (2020), 3089–3094. PMID: 32227852.

[20] HE, J., DONG, C., AND QIAO, Y. Modulating image restoration with continual levels via adaptive feature modification layers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 11056–11064.

[21] HEL-OR, Y., AND SHAKED, D. A discriminative approach for wavelet denoising. *IEEE Trans. Image Processing* (2008).

[22] HUANG, J.-B., SINGH, A., AND AHUJA, N. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 5197–5206.

[23] IOFFE, S., AND SZEGEDY, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167* (2015).

[24] JARRETT, K., KAVUKCUOGLU, K., RANZATO, M., AND LECUN, Y. What is the best multi-stage architecture for object recognition? In *2009 IEEE 12th international conference on computer vision* (2009), IEEE, pp. 2146–2153.

[25] KAKU, A., MOHAN, S., PARNANDI, A., SCHAMBRA, H., AND FERNANDEZ-GRANDA, C. Be like water: Robustness to extraneous variables via adaptive feature normalization. *arXiv preprint arXiv:2002.04019* (2020).

[26] KRULL, A., BUCHHOLZ, T.-O., AND JUG, F. Noise2void - learning denoising from single noisy images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 2124–2132.

[27] LAINE, S., KARRAS, T., LEHTINEN, J., AND AILA, T. High-quality self-supervised deep image denoising. In *Advances in Neural Information Processing Systems 32* (2019), pp. 6970–6980.

[28] LAWRENCE, E. L., LEVIN, B. D., MILLER, B. K., AND CROZIER, P. A. Approaches to exploring spatio-temporal surface dynamics in nanoparticles with in situ transmission electron microscopy. *Microscopy and Microanalysis 26*, 1 (2020), 86–94.

[29] LEE, A. B., MUMFORD, D., AND HUANG, J. Occlusion models for natural images: A statistical study of a scale-invariant dead leaves model. *International Journal of Computer Vision 41*, 1 (2001), 35–59.

[30] LEVIN, B. D., LAWRENCE, E. L., AND CROZIER, P. A. Tracking the picoscale spatial motion of atomic columns during dynamic structural change. *Ultramicroscopy 213* (2020), 112978.

[31] LUISIER, F., BLU, T., AND UNSER, M. A new sure approach to image denoising: Interscale orthonormal wavelet thresholding. *IEEE Transactions on Image Processing 16* (2007), 593–606.

[32] MARTIN, D., FOWLKES, C., TAL, D., AND MALIK, J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision* (July 2001), vol. 2, pp. 416–423.

[33] MATHERON, G. Random sets and integral geometry.

[34] METZLER, C. A., MOUSAVI, A., HECKEL, R., AND BARANIUK, R. G. Unsupervised learning with stein's unbiased risk estimator. *arXiv preprint arXiv:1805.10531* (2018).

[35] MOHAN, S., KADKHODAIE, Z., SIMONCELLI, E. P., AND FERNANDEZ-GRANDA, C. Robust and interpretable blind image denoising via bias-free convolutional neural networks. In *Proceedings of the International Conference on Learning Representations* (2020).

[36] MOHAN, S., MANZORRO, R., VINCENT, J. L., TANG, B., SHETH, D. Y., SIMONCELLI, E. P., MATTESON, D. S., CROZIER, P. A., AND FERNANDEZ-GRANDA, C. Deep denoising for scientific discovery: A case study in electron microscopy. *arXiv preprint arXiv:2010.12970* (2020).

[37] MONTINI, T., MELCHIONNA, M., MONAI, M., AND FORNASIERO, P. Fundamentals and catalytic applications of ceo2-based materials. *Chemical reviews 116*, 10 (2016), 5987–6041.

[38] MORAN, N., SCHMIDT, D., ZHONG, Y., AND COADY, P. Noisier2noise: Learning to denoise from unpaired noisy data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), pp. 12064–12072.

[39] NADO, Z., PADHY, S., SCULLEY, D., D'AMOUR, A., LAKSHMINARAYANAN, B., AND SNOEK, J. Evaluating prediction-time batch normalization for robustness under covariate shift. *arXiv preprint arXiv:2006.10963* (2020).

[40] NIE, Y., LI, L., AND WEI, Z. Recent advancements in pt and pt-free catalysts for oxygen reduction reaction. *Chemical Society Reviews 44*, 8 (2015), 2168–2201.

[41] PEREZ, E., STRUB, F., DE VRIES, H., DUMOULIN, V., AND COURVILLE, A. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2018), vol. 32.

[42] PITKOW, X. Exact feature probabilities in images with occlusion. *Journal of vision 10*, 14 (2010), 42–42.

[43] PORTILLA, J., STRELA, V., WAINWRIGHT, M. J., AND SIMONCELLI, E. P. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Trans. Image Processing 12*, 11 (2003).

[44] QUAN, Y., CHEN, M., PANG, T., AND JI, H. Self2self with dropout: Learning self-supervised denoising from single image. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), pp. 1887–1895.

[45] RAMANI, S., BLU, T., AND UNSER, M. Monte-carlo sure: A black-box optimization of regularization parameters for general denoising algorithms. *IEEE Transactions on image processing 17*, 9 (2008), 1540–1554.

[46] RAPHAN, M., AND SIMONCELLI, E. P. Optimal denoising in redundant representations. *IEEE Transactions on image processing 17*, 8 (2008), 1342–1352.

[47] RONNEBERGER, O., FISCHER, P., AND BROX, T. U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention, Springer, LNCS 9351* (2015), 234–241.

[48] SCHNEIDER, S., RUSAK, E., ECK, L., BRINGMANN, O., BRENDEL, W., AND BETHGE, M. Improving robustness against common corruptions by covariate shift adaptation. *Advances in Neural Information Processing Systems 33* (2020).

[49] SHETH, D. Y., MOHAN, S., VINCENT, J. L., MANZORRO, R., CROZIER, P. A., KHAPRA, M. M., SIMONCELLI, E. P., AND FERNANDEZ-GRANDA, C. Unsupervised deep video denoising. *arXiv preprint arXiv:2011.15045* (2020).

[50] SIMONCELLI, E. P., AND ADELSON, E. H. Noise removal via Bayesian wavelet coring. In *Proc 3rd IEEE Int'l Conf on Image Proc* (Lausanne, Sep 16-19 1996), vol. I, IEEE Sig Proc Society, pp. 379–382.

[51] SMITH, D. *CHAPTER 1: Characterization of nanomaterials using transmission electron microscopy*, 37 ed. No. 37 in RSC Nanoscience and Nanotechnology. Royal Society of Chemistry, Jan. 2015, pp. 1–29.

[52] SOLTANAYEV, S., AND CHUN, S. Y. Training and refining deep learning based denoisers without ground truth data. *arXiv preprint arXiv:1803.01314* (2018).

[53] SOLTANAYEV, S., AND CHUN, S. Y. Training deep learning based denoisers without ground truth data. In *Advances in Neural Information Processing Systems* (2018), vol. 31.

[54] SUN, G., ALEXANDROVA, A. N., AND SAUTET, P. Structural rearrangements of subnanometer cu oxide clusters govern catalytic oxidation. *ACS Catalysis 10*, 9 (2020), 5309–5317.

[55] TAO, F., AND CROZIER, P. Atomic-scale observations of catalyst structures under reaction conditions and during catalysis. *Chemical Reviews 116*, 6 (Mar. 2016), 3487–3539.

[56] VAKSMAN, G., ELAD, M., AND MILANFAR, P. Lidia: Lightweight learned image denoising with instance adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops* (2020), pp. 524–525.

[57] VINCENT, J. L., MANZORRO, R., MOHAN, S., TANG, B., SHETH, D. Y., SIMONCELLI, E. P., MATTESON, D. S., FERNANDEZ-GRANDA, C., AND CROZIER, P. A. Developing and evaluating deep neural network-based denoising for nanoparticle tem images with ultra-low signal-to-noise.

[58] WANG, D., SHELHAMER, E., LIU, S., OLSHAUSEN, B., DARRELL, T., BERKELEY, U., AND RESEARCH, A. tent: fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations* (2021), vol. 4, p. 6.

[59] XIE, Y., WANG, Z., AND JI, S. Noise2same: Optimizing a self-supervised bound for image denoising. *Advances in Neural Information Processing Systems 33* (2020).

[60] XU, J., HUANG, Y., CHENG, M.-M., LIU, L., ZHU, F., XU, Z., AND SHAO, L. Noisy-as-clean: Learning self-supervised denoising from corrupted image. *IEEE Transactions on Image Processing 29* (2020), 9316–9329.

[61] YOSINSKI, J., CLUNE, J., BENGIO, Y., AND LIPSON, H. How transferable are features in deep neural networks? *arXiv preprint arXiv:1411.1792* (2014).

[62] YU, W., POROSOFF, M. D., AND CHEN, J. G. Review of pt-based bimetallic catalysis: from model surfaces to supported catalysts. *Chemical reviews 112*, 11 (2012), 5780–5817.

[63] ZHANG, K., ZUO, W., CHEN, Y., MENG, D., AND ZHANG, L. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing* (2017), 3142–3155.

[64] ZHANG, K., ZUO, W., AND ZHANG, L. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing 27*, 9 (2018), 4608–4622.

[65] ZHANG, X., LU, Y., LIU, J., AND DONG, B. Dynamically unfolding recurrent restorer: A moving endpoint control method for image restoration. *arXiv preprint arXiv:1805.07709* (2018).

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] All claims are supported by extensive empirical experiments
   (b) Did you describe the limitations of your work? [Yes] see Section 7
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] see Section 8
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We include the main code. Main data and models are public.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] see Section SM 1, SM 2, SM 3
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] We report histogram of improvement in the main paper, and box plot and raw data in supplementary material, instead of just summary statistics.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] see Section SM 3
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes] see Section SM 2
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] we include our code
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]