arXiv:2005.06001v1 [eess.IV] 12 May 2020

Deep Learning Techniques for Inverse Problems in Imaging

Gregory Ongie*, Ajil Jalal†, Christopher A. Metzler‡
Richard G. Baraniuk§, Alexandros G. Dimakis¶, Rebecca Willett‖

April 2020

Abstract

Recent work in machine learning shows that deep neural networks can be used to solve a wide variety of inverse problems arising in computational imaging. We explore the central prevailing themes of this emerging area and present a taxonomy that can be used to categorize different problems and reconstruction methods. Our taxonomy is organized along two central axes: (1) whether or not a forward model is known and to what extent it is used in training and testing, and (2) whether or not the learning is supervised or unsupervised, i.e., whether or not the training relies on access to matched ground truth image and measurement pairs. We also discuss the tradeoffs associated with these different reconstruction approaches, caveats and common failure modes, plus open problems and avenues for future work.

1 Introduction

This paper concerns *inverse problems*, i.e., reconstructing an unknown signal, image, or multi-dimensional volume from observations. The observations are obtained from the unknown data by a *forward process*, which is typically non-invertible. Numerous imaging tasks fit under this framework, including image deblurring, deconvolution, inpainting, compressed sensing, superresolution, and many more. These forward processes are ill-posed, so reconstructing a *unique* solution that fits the observations is difficult or impossible without some prior knowledge about the data. Traditional methods minimize a cost function that consists of a data-fit term, which measures how well the reconstructed image matches the observations, and a regularizer, which reflects prior knowledge and promotes images with desirable properties like smoothness. Deep learning techniques are currently transforming image reconstruction methods and impact applications ranging

*Department of Statistics, University of Chicago, Chicago, IL.
†Department of Electrical and Computer Engineering, University of Texas at Austin, Austin, TX.
‡Department of Electrical Engineering, Stanford University, Stanford, CA.
§Department of Electrical and Computer Engineering, Rice University, Houston, TX.
¶Department of Electrical and Computer Engineering, University of Texas at Austin, Austin, TX.
‖Departments of Statistics and Computer Science, University of Chicago, Chicago, IL.

from geophysical, scientific and medical imaging. We provide an overview of this rapidly evolving landscape.

To be more precise, we consider inverse problems in which an unknown n-pixel image (in vectorized form) $\boldsymbol{x}^\star \in \mathbb{R}^n$ (or $\mathbb{C}^n$) is observed via m noisy measurements $\boldsymbol{y} \in \mathbb{R}^m$ (or $\mathbb{C}^m$) according to the model

$$\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}^\star) + \boldsymbol{\varepsilon},$$

where $\mathcal{A}$ is the (possibly nonlinear) forward measurement operator and $\boldsymbol{\varepsilon}$ represents a vector of noise. The goal is to recover $\boldsymbol{x}^\star$ from $\boldsymbol{y}$. More generally, we can consider non-additive noise models of the form

$$\boldsymbol{y} = \mathcal{N}(\mathcal{A}(\boldsymbol{x}^\star)),$$

where $\mathcal{N}(\cdot)$ samples from a noisy distribution. Without loss of generality, we assume that $\boldsymbol{y}, \boldsymbol{x}^\star, \mathcal{A}$, are real-valued, since most techniques presented in this paper can be generalized to complex-valued images/measurements by concatenating real and imaginary parts.

This general model is used throughout computational imaging [1], from basic restoration tasks like deblurring, super-resolution, and image inpainting [2], to a wide variety of tomographic imaging applications, including magnetic resonance imaging [3], X-ray computed tomography [4], and radar imaging [5]. The task of estimating $\boldsymbol{x}^\star$ from $\boldsymbol{y}$ is often referred to as *image reconstruction*. Classical image reconstruction methods assume some prior knowledge about $\boldsymbol{x}^\star$ such as smoothness [6], sparsity in some dictionary or basis [7–10], or other geometric properties [11–15]. Reconstruction amounts to finding an $\hat{\boldsymbol{x}}$ that is both a good fit to the observations $\boldsymbol{y}$ and also likely given the prior knowledge. A *regularization function* $r(\boldsymbol{x})$ measures the lack of conformity of $\boldsymbol{x}$ to a prior model and $\hat{\boldsymbol{x}}$ is selected so that $r(\hat{\boldsymbol{x}})$ is as small as possible while still fitting the observed data.

Recent work in machine learning has demonstrated that deep neural networks can leverage large collections of training data to directly compute regularized reconstructions across a host of computational imaging tasks [16–24]. In parallel, work on unsupervised methods showed how deep generative models can regularize by constraining the reconstructed image $\hat{\boldsymbol{x}}$ to remain on a learned manifold [25]. We explore the central prevailing themes of this emerging area and present a taxonomy that can be used to categorize different problems and reconstruction methods. We also discuss the tradeoffs associated with different reconstruction approaches and describe avenues for future work.

2 Opportunities and Recent Progress

In the last five years deep learning has demonstrated enormous potential for solving a host of imaging inverse problems; see, e.g., [26]. However, a fundamental understanding of the applicability of deep learning methods and their limitations remains in its infancy. This creates opportunity for additional research, careful scientific evaluation, and foundational understanding.

Medical imaging. Reconstructing images from projective measurements arises in MRI, CT, PET, SPECT, and many other modalities. Classical methods perform well but can be computationally demanding. Recent work on using training data to improve the reconstruction process can lead to

better image quality and orders of magnitude faster reconstructions than classical iterative methods. The potential gains are illustrated in Figure 1 and an excellent overview was recently published [27]. GE's "TrueFidelity" deep learning image reconstruction for CT imaging [28] has FDA approval as of April 2019 [29]. Nevertheless, numerous open questions remain, as described in Section 6.

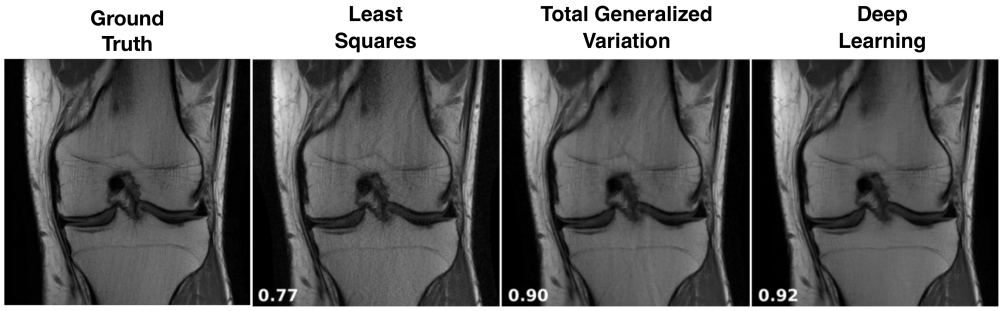


Figure 1: **Undersampled MRI reconstruction.** Undersampled multi-coil MRI reconstruction using least-squares reconstruction, total generalized variation regularization, and a deep learning approach. SSIM quality measures are displayed in white. This example highlights the potential of learned methods of image reconstruction over more conventional techniques. Figures adapted from [27].

Computational photography. The goal of computational photography is to create visually appealing images that are reasonably faithful to the scenes they represent. These conditions make deep learning an excellent candidate for computational photography reconstruction problems. For example, deep learning enables exceptional low-light imaging [30], as shown in Figure 3. Deep learning also enables estimating the depths of different objects in a scence from a photograph [31], as illustrated in Figure 4. Presently, deep learning is used to perform white balancing in the production version of Google's latest smart phone imaging systems [32, 33].

Computational microscopy. With the growing popularity of computational techniques like ptychography, solving a reconstruction problem has become an integral part of microscopy. Accordingly, there has been a surge of interest in applying deep learning to microscopy, producing new techniques to both reconstruct images and design a microscope's illumination patterns and optical elements [34–40].

Geophysical imaging. Seismic inversion and imaging involves reconstructing the Earth's interior by modeling the physical propagation of seismic waves. The comparison of simulated synthetic measurements to actual acoustic recordings of reflected waves can be used to tune the models of these ill-posed inverse problems. Deep learning techniques have been recently proposed to tackle these problems [41, 42], including methods that rely on generative models [43] constrained by partial differential equations; see also [44].

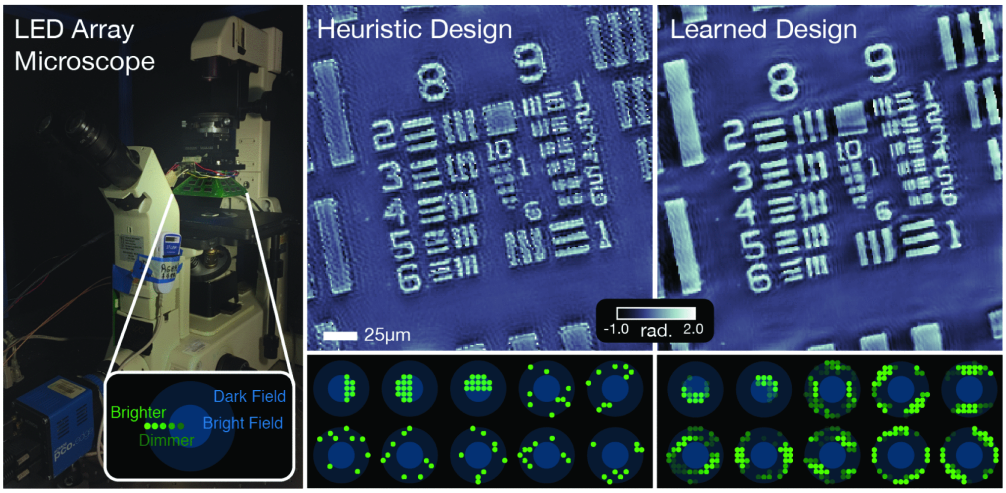
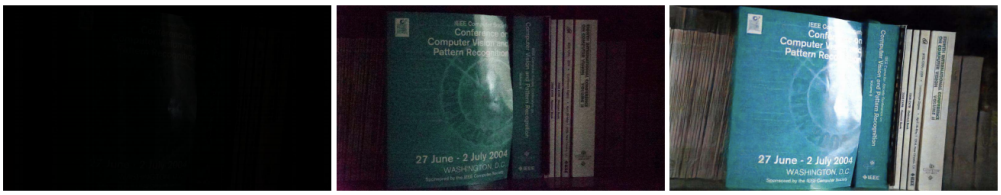


Figure 2: **Microscopy with learned illumination patterns.** Differentiable optical models and reconstruction algorithms are used to design illumination patterns (bottom row) for Fourier ptychographic microscopy. Figures adapted from [40].



(a) Camera output with ISO 8,000 (b) Camera output with ISO 409,600 (c) Our result from the raw data of (a)

Figure 3: **Low-light imaging.** A network takes an underexposed image (left) and denoises and white balances it to produce a clean image (right) that does not exhibit the color bias associated with extreme ISO images (middle). Figures adapted from [30].

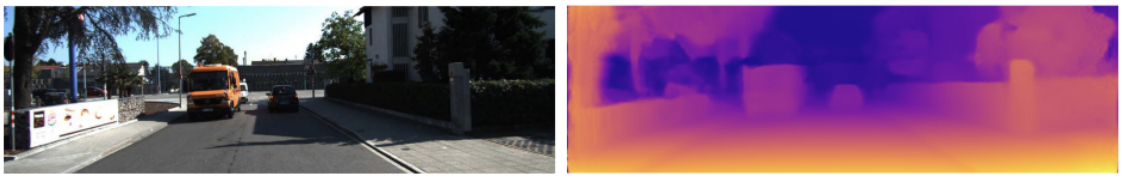


Figure 4: **Monocular depth estimation.** A network takes an image (left) and produces a depth map (right). Figures adapted from [31].

Other computational imaging applications. While still in the development stages, deep learning has shown immense promise in many other challenging computational inverse problems, including lensless imaging [45], holography [46,47], ghost imaging [48], imaging through scattering media [49–51], and non-line-of-sight imaging [52], which is illustrated in Figure 5. See [53] for a recent optics-focused review article.

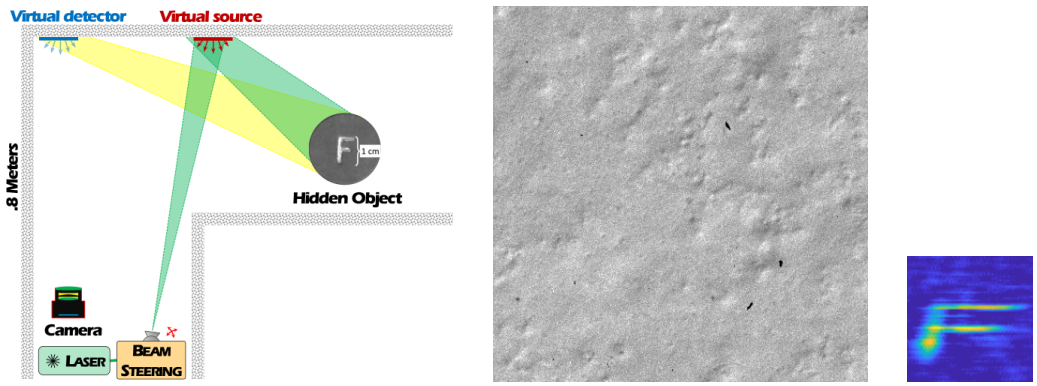


Figure 5: **Imaging around corners.** In deep-inverse correlography, deep learning is used to reconstruct a hidden object (right) from a series of speckle images (middle). Figures adapted from [52].

3 Anatomy of an Inverse Problem

Recall that we are interested in recovering a vectorized image $\boldsymbol{x} \in \mathbb{R}^n$ from measurements $\boldsymbol{y} \in \mathbb{R}^m$ of the form $\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}) + \boldsymbol{\varepsilon}$, where $\mathcal{A}$ is the (possibly nonlinear) forward measurement operator and $\boldsymbol{\varepsilon}$ represents noise. Alternatively, the system can be represented by $\boldsymbol{y} = \mathcal{N}(\mathcal{A}(\boldsymbol{x}))$, where $\mathcal{N}(\cdot)$ samples from a noisy distribution.

If the distribution of the noise is known, solving a maximum likelihood (ML) estimation problem can recover $\boldsymbol{x}$:

$$\hat{\boldsymbol{x}}_{\text{ML}} = \arg\max_{\boldsymbol{x}} p(\boldsymbol{y}|\boldsymbol{x}) = \arg\min_{\boldsymbol{x}} -\log p(\boldsymbol{y}|\boldsymbol{x}),$$

where $p(\boldsymbol{y}|\boldsymbol{x})$ is the likelihood of observing $\boldsymbol{y}$ if $\boldsymbol{x}$ were the true underlying image. (Knowledge of $\mathcal{A}$ or integration over a distribution of possible $\mathcal{A}$'s is implied in this formulation.) The maximum likelihood approach has some significant drawbacks, including potentially non-unique solutions (*e.g.,* when $\mathcal{A}$ is a linear operator with rank less than n) or high sensitivity to noise (*e.g.,* when the spectrum of $\mathcal{A}$ is not bounded below; in the case where $\mathcal{A}$ is the linear operator A, this corresponds to some eigenvalues of $A^\top A$ being small).

In some settings, one might have prior knowledge about which $\boldsymbol{x}$ are more likely; for instance, we might expect $\boldsymbol{x}$ to be smooth, or be smooth away from edges and boundaries. Such knowledge can be codified into a prior distribution for $\boldsymbol{x}$, leading to a *maximum a posteriori* (MAP) estimate

$$\hat{\boldsymbol{x}}_{\text{MAP}} = \arg\max_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{y}) = \arg\max_{\boldsymbol{x}} p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x}) = \arg\min_{\boldsymbol{x}} -\ln p(\boldsymbol{y}|\boldsymbol{x}) - \ln p(\boldsymbol{x}).$$

For the special case of additive white Gaussian noise, the MAP formulation leads to

$$\arg\min_{\boldsymbol{x}} \tfrac{1}{2}\|\mathcal{A}(\boldsymbol{x}) - \boldsymbol{y}\|_2^2 + r(\boldsymbol{x}), \tag{1}$$

where $r(\boldsymbol{x})$ is proportional to the negative log-prior of $\boldsymbol{x}$. Examples of this framework include Tikhonov regularization [54], sparsity regularization in some basis or frame [55, 56], and total variation regularization [11, 57]. In some settings, MAP estimation with underdetermined $\mathcal{A}(\cdot)$ can be considered an algorithmic procedure for choosing, among the infinitely many values of $\boldsymbol{x}$ that satisfy $y = \mathcal{A}(\boldsymbol{x})$, the one that is most likely under the prior.

While in principle MAP estimation can be used to solve most image reconstruction problems, difficulties arise when (1) the statistics of the noise are not known, (2) the distribution of the signal is not known or the log-likelihood does not have a closed form, or (3) the forward operator is not known or only partially known. In the last five years, machine learning has provided machinery to (partially) overcome many of these issues. Variations on the aforementioned inverse problem appear in a range of imaging settings. We highlight a few prominent examples in Table 1.

Table 1: *Examples of inverse problems in imaging*

Application	Forward model	Notes
Denoising [58]	$A = I$	I is the identity matrix
Deconvolution [58, 59]	$\mathcal{A}(\boldsymbol{x}) = \boldsymbol{h} * \boldsymbol{x}$	$\boldsymbol{h}$ is a known blur kernel and $*$ denotes convolution. When $\boldsymbol{h}$ is unknown the reconstruction problem is known as blind deconvolution.
Superresolution [60, 61]	$A = SB$	S is a subsampling operator (identity matrix with missing rows) and B is a blurring operator cooresponding to convolution with a blur kernel
Inpainting [62]	$A = S$	S is a diagonal matrix where $S_{i,i} = 1$ for the pixels that are sampled and $S_{i,i} = 0$ for the pixels that are not.
Compressive Sensing [63, 64]	$A = SF$ or $A =$ Gaussian or Bernoulli ensemble	S is a subsampling operator (identity matrix with missing rows) and F discrete Fourier transform matrix.
MRI [3]	$A = SFD$	S is a subsampling operator (identity matrix with missing rows), F is the discrete Fourier transform matrix, and D is a diagonal matrix representing a spatial domain multiplication with the coil sensitivity map (assuming a single coil aquisition with Cartesian sampling in a SENSE framework [65]).
Computed tomography [58]	$A = R$	R is the discrete Radon transform [66].
Phase Retrieval [67–70]	$\mathcal{A}(\boldsymbol{x}) = \lvert A\boldsymbol{x}\rvert^2$	$\lvert\cdot\rvert$ denotes the absolute value, the square is taken elementwise, and A is a (potentially complex-valued) measurement matrix that depends on the application. The measurement matrix A is often a variation on a discrete Fourier transform matrix.

3.1 Supervised vs. Unsupervised Inversion

We start by explaining a central dichotomy in the literature and in our proposed taxonomy of approaches to inverse problems. The first (and most well-known) family of deep learning inversion methods use what we call **supervised** inversions. The central idea is to create a *matched* dataset

of ground truth images $\boldsymbol{x}$ and corresponding measurements $\boldsymbol{y}$, which can be done by simulating (or physically implementing) the forward operator on clean data, i.e. measure them. Subsequently, one can train a network that takes in measurements $\boldsymbol{y}$ and reconstructs the image $\boldsymbol{x}$, i.e. learns an inverse mapping. Such supervised methods typically perform very well, but are sensitive to changes or uncertainty to the forward operator $\mathcal{A}$. In addition, a new network needs to be trained every time the measurement process changes.

The second family of techniques we cover are **unsupervised**, i.e. do not rely on a matched dataset of images $\boldsymbol{x}$ and measurements $\boldsymbol{y}$. In our taxonomy we separate unsupervised methods into three different kinds: (1) methods which use unpaired ground truth images $\boldsymbol{x}$'s and measurements $\boldsymbol{y}$'s, (2) methods which leverage ground truth images $\boldsymbol{x}$'s only, and (3) methods which use only measurements $\boldsymbol{y}$'s.

3.2 Background on Deep Generative Models

A central challenge in the foundations of learning is succinctly modeling high-dimensional distributions in a way that permits efficient learning and inference. Simply put, the difficulty is that representing a general joint probability distribution over n variables, even for binary random variables, requires $2^n - 1$ parameters. Therefore, we must postulate some type of *structure* on the data to overcome its worst-case complexity.

Previous efforts towards this goal have been at the heart of substantial breakthroughs. For example, in compressed sensing and high-dimensional statistics, the notion of sparsity and low-rank are key structural assumptions in many prior works. Sparsity (e.g., in Discrete Cosine Transform (DCT) or wavelet domains) plays *the central role* in most lossy compression standards like JPEG [71], JPEG-2000 [72] and MPEG [73]. Another successful example is graphical models: in this case a high dimensional distribution becomes tractable through factorization, which is equivalent to conditional independence. For Bayesian networks and undirected graphical models there is a rich theory for both learning and inference including precise conditions under which structure learning can be achieved efficiently.

Here, we are interested in a different way of modeling high-dimensional distributions: *Deep Generative Models* (DGMs). DGMs represent a complex distribution using a deterministic transformation applied to a simple "seed" distribution (e.g., independent Gaussian). Formally, a DGM is described by a function $G : \mathbb{R}^k \to \mathbb{R}^n$ parametrized by a deep neural network (typically convolutional) that is trained from actual data in an unsupervised way. Two primary DGM examples are Variational Auto-Encoders (VAEs) [74] and Generative Adversarial Networks (GANs) [75]. DGMs are demonstrating unprecedented visual results for image generation, but many central theoretical questions about them remain poorly understood. We will discuss how DGMs can be used as priors for inverse problems [25, 76, 77]. Figure 6 provides an illustrative example of a deep generative model that maps from $\mathbb{R} \to \mathbb{R}^2$.

4 Taxonomy for Learning to Solve Inverse Problems

In recent years there has been an emerging body of literature on using training data to solve inverse problems in imaging. Some methods use the MAP formulation, seen in Equation (1) as a starting point and attempt to learn the regularizer r or some functional of r, while others attempt to directly

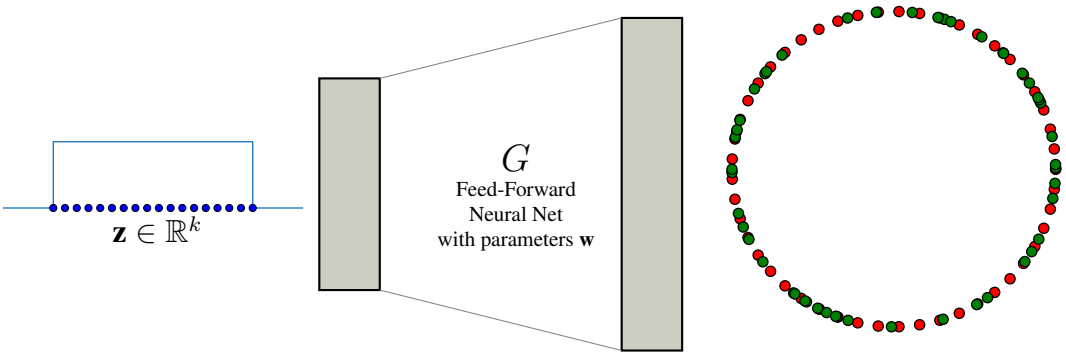


Figure 6: A deep generative model is a function $G(\mathbf{z})$ that takes a low-dimensional random vector $\mathbf{z}$ in $\mathbb{R}^k$ and produces a high-dimensional sample $G(\mathbf{z}) \in \mathbb{R}^n$. In the example shown in the figure, the generator $G : \mathbb{R} \to \mathbb{R}^2$ learns to map low-dimensional samples $\mathbf{z}$ (blue dots) drawn from a uniform distribution (blue line), such that the distribution of $G(\mathbf{z})$ (green dots) resembles the distribution of training samples (red dots). While the output of this simple generative model lies in $\mathbb{R}^2$, modern deep generative models are capable of generating images with millions of pixels. The low dimensional samples are typically drawn from a distribution which is easy to sample from, such as i.i.d. Gaussian or uniform distributions. Furthermore, the function is usually a convolutional neural network and is therefore continuous and differentiable almost everywhere.

learn a mapping from measurements $\boldsymbol{y}$ to images $\hat{\boldsymbol{x}}$. In this section, we describe a taxonomy for these approaches that facilitates an easier comparison among different methods and a better understanding of the tradeoffs among them. The overall taxonomy is shown in Table 2 and details are described below.

Computational imaging techniques often rely on a *forward model* $\mathcal{A}$, *i.e.,* a computational model of the physics underlying the measurement process. A key distinction between many types of learned inverse problem solvers is what is known when about the forward model. Options include:

- $\mathcal{A}$ is known from the beginning (*i.e.,* even during the training process). Examples include the discrete Radon and X-ray transforms in computed tomography, and the discrete Fourier transform in magnetic resonance imaging.
- $\mathcal{A}$ is not known during training, but after training may be used at test time (*i.e.,* during the reconstruction process). This framework is useful for training a general purpose model that can be "plugged in" to a variety of reconstruction tasks.
- $\mathcal{A}$ is partially known. For instance, it might rely on calibration parameters that are unknown or difficult to estimate, as is the case in blind deconvolution problems encountered in optical imaging.
- $\mathcal{A}$ is never known or modeled. In this case, all information about $\mathcal{A}$ is represented in the training data.

Each of these settings requires different methods and analyses; we elaborate below. If an accurate forward model is known – even partially – then one might argue it should be used during training

so that parameters are not wasted on "learning the physics". Indeed, several studies show that making effective use of forward models in training and testing dramatically reduces the sample complexity of learning-based image reconstruction. However, even if the forward model is known, it may be computationally prohibitive to apply. This becomes especially problematic in the training phase, where each backpropagation step may require multiple applications of the forward model or its adjoint. In this case, reconstruction architectures need to be carefully designed to reduce the number of applications the forward model or its adjoint.

Table 2: *Major categories of methods learning to solve inverse problems based on what is known about the forward model $\mathcal{A}$ and the nature of the training data, with examples for each. Details are described throughout Section 4.*

	Supervised with matched (x, y) pairs	**Train from unpaired x's and y's (Unpaired ground truths and Measurements)**	**Train from x's only (Ground truth only)**	**Train from y's only (Measurements only)**
$\mathcal{A}$ fully known during training and testing (§4.1)	§4.1.1: Denoising auto-encoders [16], U-Net [78], Deep convolutional framelets [79] Unrolled optimization [80–83], Neumann networks [84]	*amounts to training from $(\boldsymbol{x}, \boldsymbol{y})$ pairs*	*amounts to training from $(\boldsymbol{x}, \boldsymbol{y})$ pairs*	§4.1.2: SURE LDAMP [85, 86], Deep Basis Pursuit [87]
$\mathcal{A}$ known only at test time (§4.2)	§4.2.2	§4.2.2	§4.2.1: CSGM [25], LDAMP [88], OneNet [22], Plug-and-play [89], RED [90]	§4.2.2
$\mathcal{A}$ partially known (§4.3)	§4.3.1	§4.3.2: CycleGAN [91]	§4.3.3: Blind deconvolution with GAN's [92–94]	§4.3.4: AmbientGAN [76], Noise2Noise [95], UAIR [96]
$\mathcal{A}$ unknown (§4.4)	§4.4.1: AUTOMAP [97]	§4.4.2	§4.4.2	§4.4.2

4.1 Forward Model Fully Known During Training and Testing

When the forward model $\mathcal{A}$ is fully known[1], a wide variety of deep learning techniques can be employed to solve the inverse problem of interest. Here we will focus on the supervised setting where one has access to ground truth image/measurement pairs. We do not lose too much generality in our discussion by focusing on the supervised setting, since in unsupervised settings where one has access to ground truth images it is trivial to generate training pairs by applying the known forward model. However, the unsupervised setting where one only has access to (noisy) measurements

[1]Here we assume we know the "true" $\mathcal{A}$ and not an approximation; issues related to only approximately knowing $\mathcal{A}$ are discussed in Section 6

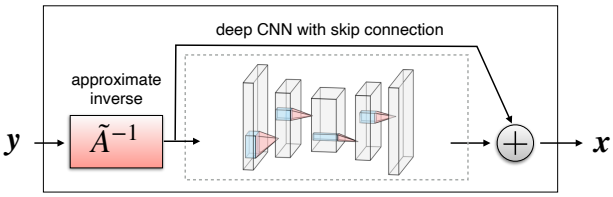


Figure 7: When an approximate inverse $\tilde{A}^{-1}$ of the forward model is known, a common approach in the supervised setting is to train a deep CNN to remove noise and artifacts from an initial reconstruction obtained by applying $\tilde{A}^{-1}$ to the measurements.

requires novel techniques, which we address below. Finally, in Section 7 we also discuss cases where one also has control over the design of $\mathcal{A}$.

4.1.1 Train from $(\boldsymbol{x}, \boldsymbol{y})$ pairs (paired ground truth and measurements)

The goal in a supervised setting is to estimate a *reconstruction network* $f_\theta(\cdot)$ that maps measurements $\boldsymbol{y}$ to images $\boldsymbol{x}$, where θ is a vector of parameters to be estimated from training data (e.g., neural network weights). Different deep learning approaches in the supervised setting can be thought of as different ways to parameterize the reconstruction network f_θ. Specifically, $\mathcal{A}$ itself (or mappings related to $\mathcal{A}$ such as adjoints or derivatives) can be embedding into the architecture defining f_θ. For simplicity, below we will assume $\mathcal{A}$ is a linear operator, denoted by A, though many of the approaches we discuss extend naturally to nonlinear operators, as well.

One simple method of incoroporating knowledge of A into the reconstruction network is by applying an approximate inverse of A, which we denote by $\widetilde{A}^{-1}$ (i.e., a matrix such that $\widetilde{A}^{-1}A\boldsymbol{x} \approx \boldsymbol{x}$ for all images $\boldsymbol{x}$ of interest), to first map the measurements back to image domain and then train a neural network to remove artifacts from the resulting images. The specific choice of $\widetilde{A}^{-1}$ will depend on the particular inverse problem, but common choices include the adjoint $A^\top$ or psuedo-inverse $A^\dagger$, though one is not limited to these. For example, in super-resolution a common choice of $\widetilde{A}^{-1}$ is upsampling by bicubic interpolation [18]; in CT reconstruction, a common choice of $\widetilde{A}^{-1}$ is filtered back projection [98]. This approach can be viewed as learning a reconstruction network whose first-layer weights are fixed and given by $\widetilde{A}^{-1}$. In this case, it is often beneficial to use a residual (or "skip") connections in the reconstruction network, since the output from the first layer is expected to be close to the output of the network. More precisely, this approach structures the reconstruction map f_θ as

$$f_\theta(\boldsymbol{y}) = g_\theta(\widetilde{A}^{-1}\boldsymbol{y}) + \widetilde{A}^{-1}\boldsymbol{y} \tag{2}$$

where g_θ is a trainable neural network depending on parameters θ; see Figure 7 for an illustration. In this case, the network g_θ is interpreted as predicting the residual between the approximate inverse and the reconstructed image. For example, in a super-resolution context, the network g_θ is predicts the missing high frequency content from a low-pass-filtered image. Networks with more complicated hierarchical skip connections are also commonly used, including the U-net [99] and architectures inspired by wavelet decompositions [79].

Inspired by iterative optimization approaches, *unrolled* methods go further and incorporate A into multiple layers of the reconstruction network, as illustrated in Figure 8. To motivate this approach, consider the MAP formulation (1) where the regularizer $r(\cdot)$ (or, equivalently, the nega-

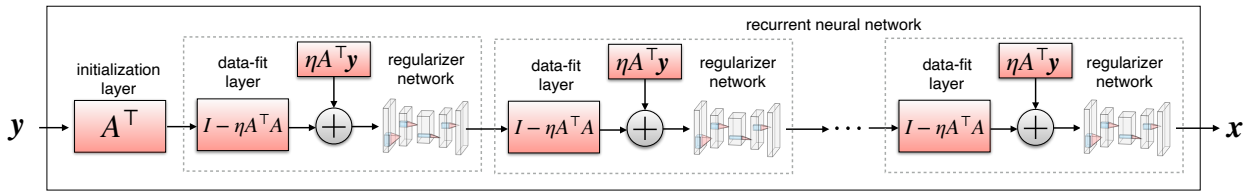


Figure 8: If the forward model A and its adjoint $A^\top$ are known, then an iterative or unrolled network embeds A and $A^\top$ in multiple layers of the network using a recurrent block artitecture. Here this approach is illustrated with an unrolling of the proximal gradient descent algorithm, where the proximal map is replaced with a deep CNN.

tive log-prior) is convex. A commonly-used algorithm for optimizing (1) in this case is proximal gradient descent [100], whose iterations have the form:

$$\boldsymbol{x}^{(k+1)} = P\left(\boldsymbol{x}^{(k)} - \eta A^\top (A\boldsymbol{x}^{(k)} - \boldsymbol{y})\right) \tag{3}$$

where $P(\boldsymbol{z}) := \arg\min_{\boldsymbol{x}}\{\frac{1}{2}\|\boldsymbol{x}-\boldsymbol{z}\|^2 + r(\boldsymbol{x})\}$ denotes the proximal operator corresponding to the regularizer $r(\cdot)$, and η is a step-size parameter. Suppose that we take as our reconstruction network the Kth iterate of proximal gradient descent $\boldsymbol{x}^{(K)}$ starting from the initialization $\boldsymbol{x}^{(0)} = 0$. Then we can turn this into a trainable reconstruction network by replacing all instances of the proximal operator $P(\cdot)$ with a trainable deep CNN $P_\theta(\cdot)$ mapping from images to images. In this approach the reconstruction network can be interpreted as learning a proximal operator. Any other free parameters, such as the step-size parameter η can also be learned in training.

The unrolling approach presented above was pioneered in [101] in a sparse coding context. Unrolled versions of (proximal) gradient gradient descent for inverse problems in imaging are investigated in [80, 102–104]. Other optimization algorithms that have been investigated in an unrolling context include the alternating directions method of multipliers (ADMM) [19], primal-dual methods [83], half-quadratic splitting [105, 106], block coordinate descent [107–109], alternating minimization [82], and approximate message passing [88, 110]. Beyond unrolling of optimization algorithms, recent work [84] considers an unrolling strategy based on a Neumann series approximation to the solution map of Equation (1).

4.1.2 Train from $\boldsymbol{y}$'s only (measurements only)

If both the forward model A and the noise statistics are known, then the measurements themselves can be used as a proxy for the ground truth. In this case, it is possible to train reconstruction networks similar to those in Section 4.1.1 from the measurements alone, with an approporiate modification of the training loss function. This is known as self-supervision, and has been studied in [87, 111] to learn autoencoders for estimating images $\boldsymbol{x}^\star$ from noisy measurements $\boldsymbol{y}$. Below we highlight a self-supervised approach based on Stein's Unbiased Risk Estimator (SURE).

GSURE In classical statistics, SURE [112] is a technique to compute the mean square error of a mean estimator, without access to the ground truth. In order to understand how it can be used in deep learning for inverse problems, consider the denoising problem, where $\boldsymbol{y} = \boldsymbol{x}^\star + \boldsymbol{\varepsilon}$. Given a

parametric class of estimators $\{f_\theta\}_{\theta\in\Theta}$ parameterized by $\theta \in \Theta$, SURE estimates the mean square error of f_θ given $\boldsymbol{y}$ as

$$\mathbb{E}_{\boldsymbol{\varepsilon}}\left[\frac{1}{n}\|\boldsymbol{x}^\star - f_\theta(\boldsymbol{y})\|^2\right] = \mathbb{E}_{\epsilon}\left[\frac{1}{n}\|\boldsymbol{y} - f_\theta(\boldsymbol{y})\|^2\right] + 2\frac{\sigma^2}{n}\text{div}_{\boldsymbol{y}}\left(f_\theta(\boldsymbol{y})\right) - \sigma^2,$$

where σ^2 is the variance of $\boldsymbol{\varepsilon}$ and $\text{div}_{\boldsymbol{y}}(f_\theta(\boldsymbol{y})) := \sum_{i=1}^n \frac{\partial f_\theta(\boldsymbol{y})}{\partial y_i}$. Notice that computing the right-hand side of this equation does not require knowledge of $\boldsymbol{x}^\star$.

If the estimators are differentiable with respect to θ, then we can use gradient descent to learn a good estimator (i.e., θ^* are the parameters obtained by gradient descent, then the estimate of $\boldsymbol{x}^\star$ is given by $f_{\theta^*}(\boldsymbol{y})$). This permits denoisers that are learned using noisy measurements alone. SURE can be generalized to other forward models A via GSURE [85, 113], by minimizing the following functional with respect to θ:

$$\mathbb{E}_{\varepsilon}\left[\frac{1}{n}\|P_A(\boldsymbol{x}^\star - f_\theta(\boldsymbol{y}))\|^2\right] = \mathbb{E}_{\varepsilon}\left[\frac{1}{n}\|P_A\boldsymbol{x}^\star\|^2 + \frac{1}{n}\|P_A f_\theta(\boldsymbol{y})\|^2 - \frac{2}{n}f_\theta(\boldsymbol{y})^T A^\dagger \boldsymbol{y} + \frac{2\sigma^2}{n}\text{div}_{\boldsymbol{y}}(f_\theta(\boldsymbol{y}))\right],$$

where $A^\dagger$ is the pseudoinverse of A and $P_A = A^\dagger A$ is projection onto the row space of A.

Notice that there is considerable freedom in choosing the function f_θ. In particular, one can employ any of the reconstruction networks described in the supervised setting above. In [85, 86, 114], the authors apply SURE to train the DnCNN [21] and Learned Denoising-Based Approximate Message Passing (LDAMP) networks [88] for denoising and compressive sensing tasks.

4.2 Forward Model Known Only at Test Time

We now consider the case where the forward model $\mathcal{A}$ is known only at test time, and one has access to representative samples of the ground truth during training. The algorithms surveyed here have the property that after training a deep model once, the same deep model can be used for any forward model. This is advantageous in situations where ground truth data is abundant, but training deep models for different forward models is expensive.

4.2.1 Train from x's only (ground truth only)

When presented with only ground truth data at train time, there are two popular approaches in the literature. The first learns a proximal operator, or denoiser, that can be used in an iterative reconstruction algorithm, while the second utilizes the ground truth training images to learn a generative prior.

Learning a proximal operator or denoiser from data. The plug-and-play (PnP) [89] approach is a powerful method for solving inverse problems using exising image denoising algorithms. The high level idea behind these methods is to use denoisers, such as BM3D [115], in place of proximal operators in iterative optimization algorithms such as ADMM [116, 117]. The denoiser acts as a regularizer for the reconstruction, and ensures good reconstruction quality at each iteration of the algorithm, such that the final reconstruction matches the measurements and satisfies the prior defined by the denoiser. A closely-related approach [90] called Regularization by Denoising (RED), proposed a general framework for PnP methods that can use deep neural networks as denoisers,

by changing the functional used for regularization. Improved methods for training deep neural networks for PnP can be found in [81, 106, 118].

Inspired by the success of Approximate Message Passing algorithms (AMP) [119] for compressed sensing, Learned Denoising-Based Approximate Messaging Passing (LDAMP) [88] learns a denoiser which can be used in a variant of AMP. Emprical results show that [88] can achieve state of the art reconstructions with a 100-fold speed improvement over other state of the art methods. Additionally, LDAMP has a state evolution heuristic which can predict the mean square error of the reconstruction at each iteration.

A similar approach is considered in [22], where a denoiser is learned from data via adversarial training. This denoiser is used as a proximal operator in the Alternating Direction Method of Multipliers (ADMM) algorithm [116] to estimate $\boldsymbol{x}^\star$.

Notice that all of these approaches are flexible and can be used to solve general inverse problems, since training the denoiser is independent of any fixed forward model.

Learning a generative prior from data. A complementary approach to learning a proximal operator is to learn a model which is capable of generating new images based on the training samples. Compressed Sensing using Generative Models (CSGM) [25] demonstrated how deep generative models can be used for solving inverse problems. The first step of CSGM [25] is to train a generative model $G : \mathbb{R}^k \to \mathbb{R}^n, k \ll n$, to capture the distribution of $\boldsymbol{x}$, given training data. This involves training a deep generative model, which can be trained using a varity of methods, such as adversarial training for GANs [75] or variational inference for VAEs [74]. Once a deep generative model G is trained, the estimate of a measured image $\boldsymbol{x}^\star$ is obtained by solving the following optimization problem:

$$\hat{\boldsymbol{z}} := \underset{\boldsymbol{z}\in\mathbb{R}^k}{\arg\min} \|AG(\boldsymbol{z}) - \boldsymbol{y}\|^2, \tag{4}$$

and the reconstruction is given by $G(\hat{\boldsymbol{z}})$. In words, we search in the latent space of the generative model $\mathbb{R}^k$ for a generated image that best explains the measurements. The optimization problem (4) is non-convex and actually NP-hard [120]. CSGM [25] proposed solving this problem by starting from a random initalization $z_0 \in \mathbb{R}^k$ and performing gradient descent (or ADAM [121]) to find the generated image that best fits the measurements.

Similar ideas like projections on smooth manifolds and additional structure beyond sparsity in inverse problems have been studied in earlier signal processing work, e.g. [122–125]. Empirical results in [25] show that CSGM can achieve similar reconstruction quality using $5 - 10$ fold fewer measurements compared to sparsity-based LASSO methods.

CSGM [25] also generalized the theoretical framework of compressive sensing and restricted eigenvalue conditions [126–129] for signals lying on the range of a deep generative model. For random subgaussian measurement matrices A, a condition called the Set Restricted Eigenvalue condition (SREC) [25], can be used to show the following two results:

- if G is an $L-$Lipschitz function, $m = O(k \log \frac{Lr}{\delta})$ measurements suffice to guarantee $\|G(\hat{\boldsymbol{z}}) - \boldsymbol{x}^\star\| \leq 6 \min_{\boldsymbol{z}:\|\boldsymbol{z}\|\leq r} \|G(\boldsymbol{z}) - \boldsymbol{x}^\star\| + \delta$.
- if G is a $d-$layered feedforward neural network with piecewise linear activation functions, then $m = O(kd \log n)$ measurements suffice to guarantee $\|G(\hat{\boldsymbol{z}}) - \boldsymbol{x}^\star\| \leq 6 \min_{\boldsymbol{z}\in\mathbb{R}^k} \|G(\boldsymbol{z}) - \boldsymbol{x}^\star\|$.

More recently, lower bounds [77, 130] established that these numbers of measurements are order optimal. Further, [77] demonstrated that deep generative models can produce all k-sparse signals, hence modeling structure with DGMs is a strict generalization of sparsity. The subgaussian assumptions on A were relaxed in [131], which further proposed a new algorithm which is robust to heavy tailed noise and arbitrary outliers. Further, asymptotically optimal results can be found in [132, 133].

These results guarantee that the optimium $\hat{\boldsymbol{z}}$ of (4) will be close to the best possible representation that the generative model can achieve. Unfortunately, actually finding this optimum is computationally hard [120] so it not known what can be achieved provably in polynomial time, despite the excellent empirical performance of gradient descent[2] Hand et al. [136] made important theoretical progress assuming that the weights of the generative model G are random and independent. For random weights, and further assuming that each layer of the generative model grows by a logarithmic factor, [136] proved that the objective (4) has only two local minima and can be optimized by gradient descent. An analysis of projected gradient descent for this problem was given by [137], while ADMM methods were proposed and analyzed in [138]. Analyzing gradient descent inversion for generative models that do not expand logarithmically per layer (as postulated by [136]) remains as an open problem. The least squares objective in (4) can be seen as a projection on the range of a generator, and was also independently proposed in [139–141].

The CSGM approach has been generalized to tackle different inverse problems, algorithms for decoding, and to other assumptions on the generative model. Examples of different inverse problems include phase retrieval [133, 142], blind deconvolution [93], geophysical seismic imaging [43], bilinear estimation [143], and 1-bit compressed sensing [144]. [145, 146] propose improvements to the objective function in (4). Alternate algorithms for decoding including ML-VAMP [147, 148] and Surfing [149]. The results in [150] provide uncertainty quantification for the reconstruction.

While trained generative models are a powerful tool for solving inverse problems, training them can be challenging since they require massive datasets and a long training time. Surprising results show that *untrained generative networks* can solve this difficulty, and we review this line of work in Section 7.

4.2.2 Train from $\boldsymbol{y}$'s only (measurements only)

Having $\boldsymbol{y}$'s among the training data suggests that there is a fixed $\mathcal{A}$ generating training samples. While work in this setting should provide few technical challenges, we are unaware of application domains in which this paired data would be available yet $\mathcal{A}$ would be wholly unknown except at test time.

4.3 Forward Model Partially Known

We now consider inverse problems where the forward operator is partially known. This can occur, for example, when the forward model is parametric and we know either the distribution of or sufficient statistics about the parameters.

[2]Empirically, gradient descent inversion works well for medium-sized generative models like DCGAN but has not been very effective in inverting bigger generators like BigGAN [134], see [135] for a discussion.

4.3.1 Train from (x, y) pairs (paired ground truth and measurements)

In general, knowledge of $\mathcal{A}$ arises from a mathematical model of an imaging system or careful calibration procedures. In either case, we typically only know an approximation of $\mathcal{A}$. In general, these inaccuracies can complicate the reconstruction process, but when we have real-world training observations of the form $(\boldsymbol{x}, \boldsymbol{y})$, then we can expect those samples to reflect the true $\mathcal{A}$. As a result, training a deep neural network to perform reconstruction can leverage the partial knowledge of $\mathcal{A}$ to perform some approximate inversion of the measurement process while using the training data to learn to remove "artifacts" and compensate for inaccuracies in the model. See the illustration in Figure 7.

4.3.2 Train from unpaired x's and y's (unpaired ground truth and measurements)

In certain cases, one has access to unpaired samples of the ground truth and measurements. That is, if $\boldsymbol{x}_i^\star, \boldsymbol{y}_i$ denote the i^{th} training sample of the ground truth and measurement, then $\boldsymbol{x}_i^\star, \boldsymbol{y}_i$ follow the marginal distributions of $\boldsymbol{x}^\star, \boldsymbol{y}$, without following the joint distribution of $(\boldsymbol{x}^\star, \boldsymbol{y})$. This can occur for example, if one has clean MRI scans as ground truth, and MRI scans with motion blur as measurements, without any pairing between the clean and blurry scans.

Models like CycleGAN [151] are well-suited for this situation, as they can learn forward and backward mappings between the image and measurement domain, given unpaired samples of images and measurements. This idea has been explored in [152], for removing motion blur from MRI scans, as well converting PET scans to CT scans. A similar idea was explored for MRI by [153], although in this case the forward operator is assumed to be a subsampled Fourier transform.

We briefly describe the original CycleGAN algorithm, as its extension to inverse problems can be derived with domain specific modifications. Let $p_{\boldsymbol{x}}, p_{\boldsymbol{y}}$, denote the distributions over $\boldsymbol{x}, \boldsymbol{y}$. CycleGAN aims to learn two generative models $F : \mathcal{X} \to \mathcal{Y}, G : \mathcal{Y} \to \mathcal{X}$, where $\mathcal{X}, \mathcal{Y}$ are respectively the image and measurement domain. Since G, F, need to be trained with unpaired $\boldsymbol{x}$'s and $\boldsymbol{y}$'s, one way to create a joint distribution between $\boldsymbol{x}, \boldsymbol{y}$ is to make G, F approximate inverses of each other. That is, for all $\boldsymbol{y} \in \mathcal{Y}, F(G(\boldsymbol{y})) \approx \boldsymbol{y}$, and for all $\boldsymbol{x} \in \mathcal{X}, G(F(\boldsymbol{x})) \approx \boldsymbol{x}$. In [151], this requirement is satisfied by introducing the *cycle consistency loss*, defined as

$$\mathcal{L}_{cyc}(G, F) = \mathbb{E}_{p_{\boldsymbol{x}}}[\|\boldsymbol{x} - G(F(\boldsymbol{x}))\|_1] + \mathbb{E}_{p_{\boldsymbol{y}}}[\|\boldsymbol{y} - F(G(\boldsymbol{y}))\|_1].$$

By adding this cycle loss to individual adversarial losses for G and F, they can be simultaneously trained. Once they are trained, F, G can be used to map from the image domain to the measurement domain, and vice versa. For example, if $\boldsymbol{y}$ is an MRI scan with motion blur, then $G(\boldsymbol{y})$ will remove the blur present in $\boldsymbol{y}$.

4.3.3 Train from x's only (ground truth only)

Generative priors [74, 75] have been successfully applied in many inverse problems [25], and are a good option when one has access to samples of $\boldsymbol{x}$. These priors have also found success in problems like blind deconvolution [93], where $\mathcal{A}(\boldsymbol{x}) = \boldsymbol{x} \circledast A$, and the distribution of A is known, but we do not know the exact blurring kernel that produced the measurements. In [93], it is assumed that there exists two generative models given by $G_A, G_{\boldsymbol{x}}$: the output distribution of G_A captures the distribution of A, whereas $G_{\boldsymbol{x}}$ captures the distribution of $\boldsymbol{x}$.

Given $G_A, G_{\boldsymbol{x}}$, and measurements $\boldsymbol{y} = \boldsymbol{x}^\star \circledast A^* + \boldsymbol{\varepsilon}$, one can recover the ground truth $\boldsymbol{x}^\star$ and the blurring kernel A^* by solving the following optimization problem:

$$\hat{\boldsymbol{z}}_{\boldsymbol{x}}, \hat{\boldsymbol{z}}_A = \underset{\boldsymbol{z}_{\boldsymbol{x}} \in \mathbb{R}^k, \boldsymbol{z}_A \in \mathbb{R}^k}{\arg\min} \|y - G_{\boldsymbol{x}}(\boldsymbol{z}_{\boldsymbol{x}}) \circledast G_A(\boldsymbol{z}_A)\|^2. \tag{5}$$

Once this is solved, the estimates for $\boldsymbol{x}^\star, A^*$, are given by $G_{\boldsymbol{x}}(\hat{\boldsymbol{z}}_{\boldsymbol{x}}), G_A(\hat{\boldsymbol{z}}_A)$.

This approach was generalized to blind demodulation by [154]. [154] also provide theoretical guarantees on the loss landscape of the above objective, and show that it can be minimized by gradient descent.

Another approach is DeblurGAN [92], in which a GAN is trained end-to-end using blurry images as input. During training, clean images are synthetically blurred, and the GAN must learn to generate crisp images given blurry images. This produces cripser images, but it can be expensive, since a minor change in the distribution of A would require retraining the GAN.

4.3.4 Train from $\boldsymbol{y}$'s only (measurements only)

Learning from measurements alone is a hard task, which is further compounded by the difficulty of only having partial knowledge of the forward model. In order to tackle this problem, it is commonly assumed that the forward operator has an underlying distribution, and we have knowledge of its statistics. There are two popular ways of solving this problem: one is a "supervised" approach, and another is via adversarial training. We first explore the supervised approach.

Noise2Noise. Noise2Noise [95] learns a neural network $f_\theta : \mathbb{R}^m \to \mathbb{R}^n$ that accepts noisy measurements as input and produces clean samples as output. The training of f is reminiscent of supervised training, except it does not actually need ground truth. In order to train f, [95] assume

- The training data consists of $(\tilde{\boldsymbol{x}}, \boldsymbol{y})$ pairs, where $\tilde{\boldsymbol{x}}$ is a noisy version of $\boldsymbol{x}^\star$, and $\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}^\star) + \boldsymbol{\varepsilon}$.
- The samples $\tilde{\boldsymbol{x}}$ satsify $\mathbb{E}[\tilde{\boldsymbol{x}}|\boldsymbol{y}] = \boldsymbol{x}^\star$.

Given this dataset, the learned neural network is f_{θ^*}, where $\theta^* = \arg\min_\theta \mathbb{E}[\|f_\theta(\boldsymbol{y}) - \tilde{\boldsymbol{x}}\|^2]$. The theoretical argument for Noise2Noise is based on the assumption $\mathbb{E}[\tilde{\boldsymbol{x}}|\boldsymbol{y}] = \boldsymbol{x}^\star$. This allows f to be trained from contaminated samples $\tilde{\boldsymbol{x}}$, without access to the clean ground truth. In theory this should require multiple $\tilde{\boldsymbol{x}}$ for each $\boldsymbol{x}^\star$, but [95] observe that one sample suffices. An important benefit of this approach is that it does not need explicit knowledge of the parameters or distribution of $\mathcal{A}$. While Noise2Noise does not need the ground truth, it still requires $\tilde{\boldsymbol{x}}$, which is a noisy proxy of $\boldsymbol{x}^\star$. We now explore an alternative approach which weakens this assumption.

Adversarial training. Adversarial training has emerged as a powerful technique for learning high dimensional distributions that are hard to describe. When $\mathcal{A}$ follows a parametric distribution, AmbientGAN [76] demonstrates how adversarial training can learn from measurements alone. With a slight abuse of notation, let $\boldsymbol{y}, \boldsymbol{x}^\star, \mathcal{A}$, respectively denote random variables associated with the measurements, ground truth, and forward model. Similarly, let $P_{\boldsymbol{y}}, P_{\boldsymbol{x}^\star}, P_{\mathcal{A}}$ denote their probability distributions. Given samples from $P_{\boldsymbol{y}}$, and assuming that it is easy to sample

parameters of the forward model, AmbientGAN learns the distribution $P_{\boldsymbol{y}}$ by optimizing the following objective:

$$\min_G \max_D \mathbb{E}_{\boldsymbol{y}} \left[\log(D(\boldsymbol{y}))\right] - \mathbb{E}_{\boldsymbol{z},\mathcal{A}} \left[\log(1 - D(\mathcal{A}(G(\boldsymbol{z}))))\right],$$

where $G : \mathbb{R}^k \to \mathbb{R}^n, D : \mathbb{R}^n \to [0, 1]$ with $k \ll n$, and $\boldsymbol{z} \in \mathbb{R}^k$ is a random latent variable which can be easily sampled, for example i.i.d. Gaussian or i.i.d. uniform. The intuition for this approach is similar to that of traditional GANs [75]. In traditional GANs, the discriminator D must learn to distinguish between the distribution $P_{\boldsymbol{x}^\star}$ and $P_{G(\boldsymbol{z})}$, whereas in AmbientGAN, the discriminator must learn to distinguish between $P_{\boldsymbol{y}}$ and $P_{\mathcal{A}G(\boldsymbol{z})}$. Under certain regularity conditions on the distributions $P_{\mathcal{A}}, P_{\boldsymbol{x}^\star}$, [76] show that the distribution $P_{\boldsymbol{x}^\star}$ can be exactly recovered.

Once the AmbientGAN is trained, it can be used for inference: for a new $\mathcal{A}, \boldsymbol{y}$, the reconstruction $\hat{\boldsymbol{x}} = G(\hat{\boldsymbol{z}})$ can be obtained by solving the constrained least squares problem (4). Note that if the ground truth distribution has been exactly learned by AmbientGAN, $\mathcal{A}$ need not follow any distributional assumptions in the inference phase.

While AmbientGAN has nice theoretical properties, it can be computationally expensive, since it requires running an optimization procedure at test time. A more direct solution is to train a network $\tilde{G}$ which accepts the measurements as input and outputs a possible reconstruction. [96] is one such approach, where the reconstruction $\tilde{G}(\boldsymbol{y})$ is ideally the MAP estimate of $\boldsymbol{x}^\star$. Similar ideas have been explored in [155], although in [155] there is no stochasticity in $\mathcal{A}$.

4.4 Unknown Forward Model

In some cases the forward model may be entirely unknown, misspecified, or computationally intractible to use in training and testing. If this is the case, then one is essentially limited to the supervised setting, *i.e.,* learning must take place with matched image and measurement pairs.

4.4.1 Train from $(\boldsymbol{x}, \boldsymbol{y})$ pairs (paired ground truth and measurements)

Assuming that one only has access to image and measurement pairs $(\boldsymbol{x}, \boldsymbol{y})$ without knowledge of the forward model the simplest approch is to treat reconstruction map $\boldsymbol{y} \mapsto \boldsymbol{x}$ as a "black box" that can be well-approximated by conventional neural network architectures with the appropriate input and output dimensions, as illustrated in Figure 9. This is the approach taken in [97], which proposed the automated transform by manifold approximation (AUTOMAP) framework. In this framework, the reconstruction network f_θ is modelled as a map between a low-dimensional "measurement manifold" $\mathcal{Y}$ and an "image manifold" $\mathcal{X}$ embedded in high-dimensional Euclidean space: $f_\theta = \phi_{\boldsymbol{x}} \circ g \circ \phi_{\boldsymbol{y}}^{-1}$, where $\phi_{\boldsymbol{y}}^{-1}$ maps Euclidean space to intrinsic coordinates in $\mathcal{Y}$, g is a diffeomorphism between $\mathcal{Y}$ and $\mathcal{X}$, and $\phi_{\boldsymbol{x}}$ maps from instrinic coordinates in $\mathcal{X}$ to Euclidean space. To approximate this idealized mapping, $\phi_{\boldsymbol{y}}^{-1}$ is then parameterized as a sequence of fully connected neural network layers, while g and $\phi_{\boldsymbol{x}}$ are parameterized as a sequence of CNN layers. Note that while the experiments in [97] used knowledge of the forward model A to generate training data from fully sampled images, in principle the approach should succeed without access to A.

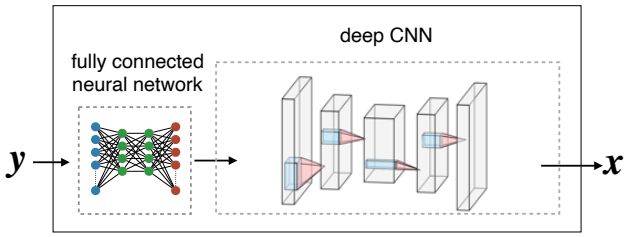


Figure 9: When the forward model A is unknown but sufficiently many training samples are available, learning a reconstruction map is still possible using deep neural networks, as proposed in the AUTOMAP framework [97].

4.4.2 Unsupervised approaches

If $\mathcal{A}$ is entirely unknown, then there are limited options without paired $(\boldsymbol{x}, \boldsymbol{y})$ training samples, as these pairs are our only mechanism for understanding anything about $\mathcal{A}$. In order to make make this problem identifiable, it is necessary to have some additional information about $\mathcal{A}$.

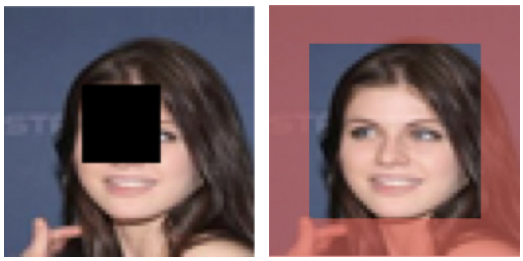
5 Key Tradeoffs

5.1 Sample Complexity vs. Generality

In many of the unsupervised learning approaches we discussed above, training takes place independently of the forward model A. This includes compressed sensing with generative models and iterative plug-and-play reconstruction with a denoising autoencoder. In these cases, one learns a generative model or denoising autoencoder using only a collection of training images, which does not require knowledge of A. The advantage of this approach is that once training has taken place, the learned generative model or denoising autoencoder can be used for *any* forward model, so we do not need to re-train a system for each new inverse problem. In other words, the learning is *decoupled* from solving the inverse problem, resulting in high generality.

However, the generality of the decoupled approach comes with a high price in terms of sample complexity. To see why, note that learning a generative model or a denoising autoencoder fundamentally amounts to estimating a full *prior* distribution over the space of images; let us denote this distribution as $p(\boldsymbol{x})$. Thoroughly understanding the space of images of interest is important if our learned regularizer is to be used for linear inverse problems of which we are unaware during training. However, when we know at training time what A is, then we only need to learn the *conditional* distribution $p(\boldsymbol{x}|\boldsymbol{y})$ where $\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{\epsilon}$. For example, consider an inpainting scenario in which we only observe a subset of pixels in the image $\boldsymbol{x}$. Rather than learn the distribution over the space of all possible images, we only need to learn the distribution over the space of missing pixels conditioned on the observed pixels, as in Figure 10. Of course, if we know the forward model A and the statistics of the noise $\boldsymbol{\epsilon}$, then $p(\boldsymbol{x}|\boldsymbol{y})$ can be calculated from $p(\boldsymbol{x})$ and A using Bayes' law. However, such an approach is not always the most sample-efficient.

For instance, imagine our images have d pixels and the distribution $p(\boldsymbol{x})$ lies in a Besov space with a smoothness parameter α, where larger α implies smoother functions that are easier to estimate [156]. Then the L^2 density estimation error scales like $N^{-\frac{\alpha}{2\alpha+d}}$, where N is the number of



(a) (b)

Figure 10: Sample complexity in learning to solve inverse problems. (a) An example of an inpainting problem in which the goal is to estimate the missing (black) pixels in the center of the image. If we do not know ahead of time which pixels may be missing, then we must learn (perhaps implicitly) the distribution over all possible images, $p(\boldsymbol{x})$. (b) If we know at training time which pixels are going to be missing, this information can be used to reduce sample complexity. For instance, if we know the missing pixels will be located as they are in (a), then the red highlighted pixels in (b) are irrelevant to the inpainting task and we do not need to learn the distribution of pixels values in those regions. Rather, we must only learn the *conditional* distirbution $p(\boldsymbol{x}|\boldsymbol{y})$, which can require significantly fewer training samples.

training samples [157–159]. In contrast, conditional density estimation errors scale like $N^{-\frac{\alpha'}{2\alpha'+d'}}$, where α' is the smoothness of the conditional density and d' is the number of pixels on which the conditional density depends [160, 161] (*i.e.,* the number of pixels not covered by the red overlay in Figure 10) . In many scenarios $\alpha' > \alpha$ and $d' \ll d$, meaning that conditional density estimation can achieve much smaller errors with many fewer training samples than strategy of first estimating the full density and then calculating the conditional density based on this higher-error estimate.

The key point is that decoupled approaches (implicitly) require learning a full prior $p(\boldsymbol{x})$ whereas a method that incorporates A into the learning process has the potential to simply learn the conditional density $p(\boldsymbol{x}|\boldsymbol{y})$, which often can be performed accurately with relatively less training data.

5.2 Reconstruction Speed vs. Accuracy

In many inverse problems in imaging, the computational bottleneck in traditional reconstruction algortihms comes from applying the forward model A (or its adjoint $A^\top$). This is a perpetual challenge in applications such as medical image reconstruction. As a result, approaches based on iterative optimization with a large number of step sizes can be quite time-consuming [11, 57, 89, 90] even though the resulting reconstruction may be highly accurate. Deep learning provides an opportunity to reduce the computational burden of model-based iterative reconstruction methods.

For instance, consider iterative reconstruction methods. Each iteration typically requires at least one application of the forward model operator $\mathcal{A}$ and its adjoint, and these calculations can be the primary computational burden of the method. Reducing the number of iterations can therefore dramatically reduce reconstrution time. In classical methods, like iterative total-variation regularized reconstruction, we have few mechanisms for controlling the number of iterations since the methods need to run to convergence to yield accurate results. However, empirical results have

shown that deep-learned based approaches can achieve comparable accuracies with far less computation. For instance, consider the unrolled optimization methods described above; there the number of blocks is akin to the number of iterations, and by fixing the number of blocks and then learning a regularizer within this framework, we essentially learn a reconstruction method that is adapted to a small number of iterations. Specialized unrolling approaches incorporating preconditioners can reduce the number of blocks and further increase the speed.

6 Caveats/Beware/Failure Modes

So far we have reviewed some of the exciting breakthroughs that have been made possible through deep learning. In this section we view these algorithms through a more critical lens, in order to understand their current limitations and failures. This raises important research questions that we must address before we can hope to apply deep learning in real world applications of inverse problems.

Robustness to different forward model at test time than at train. In some settings, the forward model used during training is different from the forward model used during testing. For example, imagine learning to reconstruct MRI images for a scanner at one clinic and then attempting to use that learned algorithm to reconstruct MRI images for a (subtly different) scanner at another clinic. The different methods described in Section 4 will have different degrees of robustness to perturbations in the forward models between training and testing. This is illustrated in Figure 11 for a few representative methods.

A related model mismatch issue arises in the discretization of forward models and learned reconstruction algorithms. For example, in order to generate training data many supervised learning methods commit the "inverse crime" [162] by assuming the forward model and ground truth images are discrete, when in fact they are defined in a continuous domain. This can lead to undesirable artifacts at test time, such as Gibb's ringing artifacts in MRI [163]. Learning-based approaches by themselves cannot resolve this issue, and need to be properly modified in order to recover artifact-free images at test time by, e.g., incorporating post-processing steps.

Recovering features not represented by training data. The central assumption underlying all machine learning based image reconstruction methods is that the training data is representative of what we might see at test time. In some applications, such as medical imaging, it is unclear to what extent that assumption holds. One might imagine patients with unusual geometries in their anatomy or tumors that are not reflected by the training set [165, 166]. The ability of learned reconstruction methods to faithfully reconstruct such features remains poorly understood and can vary from method to method, as illustrated in Figure 12.

Difficult to interpret. A side effect of the flexibility and power offered by deep learning models is that they are difficult to analyze and interpret. Hence we currently have a very poor understanding of some methods that provide state of the art results. For example, deep image prior [167] and related methods [168, 169] provide very surprising results – they do not require any training

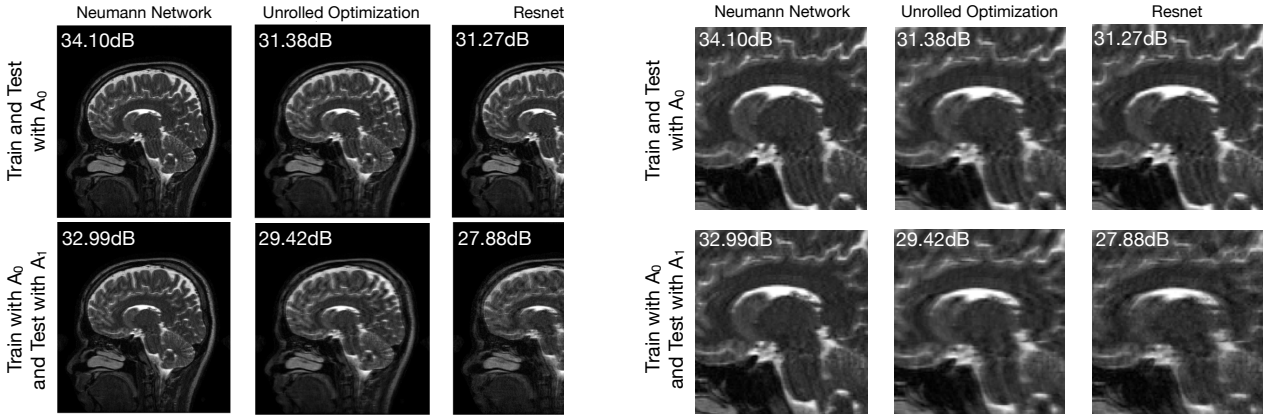


Figure 11: Robustness to forward model perturbations. (a) Reconstructions produced by Neumann networks [84], unrolled optimization [80], and a residual autoencoder [164]. The first row shows reconstructions when the true forward model is used during training and the same forward model is true and used during testing. The second row shows reconstructions when the true forward model is used during training and is also used at test time even though a different forward model generated the test data. (b) Same as (a), but zoomed into a smaller region to help show details. This example illustrates the shortcomings of learning to reconstruct for a specific forward model A and also how different learning frameworks can be more or less sensitive to perturbations of A. The tradeoffs between network architecture and robustness are not well understood and an active area of research.

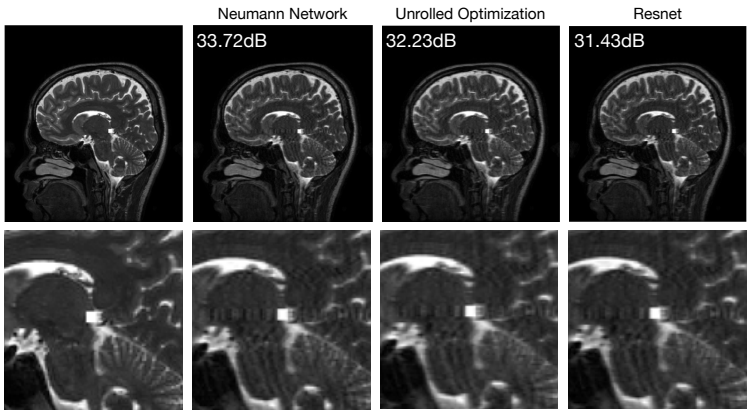


Figure 12: Robustness to image features not represented in training set. Three different reconstruction methods – Neumann networks [84], unrolled optimization [80], and a residual autoencoder [164] – were trained on an MRI training set from [82]. They are then applied to a test image that corresponds to an MRI image with a small square inserted near the center, as shown in the far left column. This type of square feature was not present in the training set. The three methods do recover the feature, but also produce non-trivial artifacts to its left and right. Both original (top) and zoomed (bottom) images are shown.

data, but are competitive with methods that leverage knowledge from large datasets. The current hypothesis for their success is that convolutional models are biased towards smooth signals, and [170] provide preliminary theoretical arguments for this hypothesis. However, a solid theoretical framework for analyzing these models remains open. A similar argument can be made against supervised models, wherein we do not understand the training phase well enough to analyze the predictions made at inference time.

Creation of artifacts. There has been significant progress in generative modeling over the last few years, and the perceptual quality of generated images is almost lifelike [134]. Early GANs struggled with images containing complicated semantic structure, but modern GANs have been able to overcome this issue. Despite the recent progress in developing better generative models, the generated images contain many artifacts and distortions. Deep learning models for medical imaging that directly map from measurements to images are also somewhat contentious. Deep learning has an incredible ability to generate realistic looking images, even when the features in the image are not actually present [166]. These artifacts can be problematic if the reconstructions are used for downstream tasks such as classification of tumors.

Failure modes may be hard to recognize. As an illustrative example, consider the case of compressed sensing using generative models [25]. The decoding algorithm always returns an image within the range of the generative model, which by design will have high perceptual quality. This is a positive when the data is well behaved and does not have outliers. Now consider a case where the forward operator is heavily underdetermined or data has outliers. In this case, the algorithm in [25] will return a high quality image even if it has failed [131]. In contrast, handcrafted algorithms Lasso would simply return a non-sparse signal. This failure mode is easy to recognize for Lasso, but if we are using a generative model, we may completely miss the failure. This prompts the need for algorithms that are either robust to outliers [131], or algorithms that can provide uncertainty quantification for their reconstructions such as in [150].

7 Open Problems and Future Directions

Control over forward model design. In many applications one has some degree over the measurement process. For instance, one can select which locations in k-space are sampled in an MRI scan or which DMD patterns are applied in a single-pixel-camera. While this problem has long been tackled using heuristics like variable density sampling [171], deep learning provides a mechanism to optimize the sampling pattern in a more principled way. One need only make the measurement model a trainable parameter that can be optimized with training data [23]. This idea can been used to do things like design illumination patterns for microscopes [38–40, 172].

Taking this idea one step further, deep learning can even be used to *design* physical systems. A series of recent works have modeled cameras as differentiable optical systems and then used back-propagation to design specialized optical filters and diffractive optical elements for improved demosaicing [173], color imaging and demosaicing [173], super-resolution and extended depth-of-field imaging [174], monocular depth estimation [175, 176], high dynamic range imaging [177], and single-lens wide-field-of-view imaging [178].

Extensions to other application domains. This tutorial has focused on inverse problems in imaging, but inverse problems abound in many different settings, including estimating boundary conditions for partial differential equations, estimating molecular structure from multi-modal measurements, radar, geophysics, and more. Many of the central themes of this tutorial, including understanding what must be known about the forward operator or the nature of the training data needed for various algorithms to be viable translate to these other settings, while the specifics of how to choose the neural network architecture or tradeoffs among different algorithms remain open questions. Furthermore, the discussion in this paper focuses on settings in which everything that is known about the physical setting of the inverse problem may be encapsulated in the forward model $\mathcal{A}$; in more general settings, in which we may have access to additional physical contraints or side information, there is an opportunity to develop frameworks for incorporating this knowledge.

Unlearned Methods. Deep Image Prior (DIP) [167] is an algorithm which uses *untrained generative models* for image reconstruction. Given measurements $\boldsymbol{y}$ and the forward operator $\mathcal{A}$, DIP initializes a generative network $G_\theta : \mathbb{R}^k \to \mathbb{R}^n$ with a fixed random input vector $\boldsymbol{z} \in \mathbb{R}^k$, and optimizes over the network weights θ. The reconstruction $\hat{\boldsymbol{x}}$ is given by $G_{\theta^*}(\boldsymbol{z})$, where

$$\theta^* = \arg\min_\theta \|\mathcal{A}(G_\theta(\boldsymbol{z})) - \boldsymbol{y}\|^2. \tag{6}$$

Experimental results in [167] show that DIP is competitive with state of the art algorithms. This result is surprising, since DIP requires no training data, and only requires measurements from a single sample. Additionally, the number of weights in the generative network often exceeds the number of pixels in the image. This implies that DIP should be able to find a set of weights that can fit any image, including Gaussian noise, and this is observed empirically in [167]. In order to avoid this, [167] uses early stopping as a regularizer. For example, if gradient descent is used to solve eq. (6), then early stopping means that gradient descent must be stopped before it converges. An intuitive explanation for the success of DIP is that convolutional neural networks are biased towards smooth, "natural" images, and hence smooth components of an image will be reconstructed before the noisier components in the measurements. Results in [169, 179, 180] generalize the results in [167].

Deep Decoder [168] is a related algorithm which fixes issues encountered in DIP. The Deep Decoder is an underparameterized network which is competitive with DIP and *does not* require early stopping. A preliminary result in [168] shows that a single layer Deep Decoder will not fit Gaussian noise, and further analysis can be found in [170]. Beyond these preliminary insights, untrained generative models remain poorly understood and theoretically surprising. A proper theoretical framework for understanding when they work, how to properly regularize them and how to measure their complexity remain significant open problems.

Transfer learning. In many settings, such as medical imaging, we may have only limited quantities of training data; in other settings, such as astronomy, we may not have access to any "real" training images but can generate simulated training data. In these settings, we face the challenge of leveraging data from a different application domain or from simulations to improve inverse problem solvers in our target domain. This challenge is generally referred to as "transfer learning" or "domain adaptation" [181, 182]. The limited emprical work today in transfer learning for inverse problems in imaging is promising and suggests the need for additional study.

Nonlinear inverse problems. Nearly all the examples presented in this tutorial had linear operators for the forward model. However, in many applications the true forward model is nonlinear. A number of works have investigated the phase retrieval problem [45, 46, 142, 180, 183–185], often with great empirical success. However, little is known about how we should solve nonlinear inverse problems generally, or the inherent tradeoffs associated with nonlinear forward models.

Uncertainty quantification. Characterizing uncertainty in solutions to an inverse problem is essential for many imaging tasks, including medical diagnosis from CT or MR images. However, most learning approaches investigated in this work do not provide uncertainty estimates. Taking a Bayesian perspective, recent work [186, 187] addresses this shortcoming by estimating a full posterior distribution of images fitting a given set of measurements (or estimate statistics derived from the posterior) using an generative adversarial training framework. This is used in [186] to give pixel-wise variance estimates and perform hypothesis testing in a CT reconstruction setting. Incorporating similar uncertainty quantification into other learning-based approaches, especially in cases where less is known about the distribution of ground truth images, is an interesting open problem.

Acknowledgments

We thank Davis Gilton for performing the MRI reconstruction experiments pictured in Figures 11 and 12 and for his helpful feedback. R.W. and G.O. were supported in part by AFOSR FA9550-18-1-0166, DOE DE-AC02-06CH11357, NSF OAC-1934637 and NSF DMS-1930049. C.M. was supported by an ORISE Intelligence Community Postdoctoral Fellowship. A.G.D. and A.J. were supported in part by NSF CCF-1618689, DMS-1723052, CCF 1763702, AF 1901292, Western Digital and the Fluor Centennial Teaching Fellowship. R.B. was supported in part by NSF CCF-1911094, IIS-1838177, IIS-1730574; ONR N00014-18-12571 and N00014-17-1-2551; AFOSR FA9550-18-1-0478; and a Vannevar Bush Faculty Fellowship, ONR N00014-18-1-2047.

References

[1] H. H. Barrett and K. J. Myers, *Foundations of image science*. John Wiley & Sons, 2013.

[2] R. C. Gonzalez and R. E. Woods, *Digital image processing*, 3rd ed. Pearson, 2007.

[3] J. A. Fessler, "Model-based image reconstruction for MRI," *IEEE Signal Processing Magazine*, vol. 27, no. 4, pp. 81–89, 2010.

[4] I. Elbakri and J. Fessler, "Statistical image reconstruction for polyenergetic X-ray computed tomography," *IEEE Transactions on Medical Imaging*, vol. 21, no. 2, pp. 89–99, 2002.

[5] R. E. Blahut, *Theory of remote image formation*. Cambridge University Press, 2004.

[6] A. N. Tychonoff and V. Arsenin, "Solution of ill-posed problems," *Winston & Sons, Washington*, 1977.

[7] M. A. Figueiredo and R. D. Nowak, "A bound optimization approach to wavelet-based image deconvolution." in *IEEE International Conference on Image Processing (ICIP)*, 2005, pp. 782–785.

[8] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *International Conference on Machine Learning (ICML)*. ACM, 2009, pp. 689–696.

[9] G. Yu, G. Sapiro, and S. Mallat, "Solving inverse problems with piecewise linear estimators: From Gaussian mixture models to structured sparsity," *IEEE Transactions on Image Processing*, vol. 21, no. 5, pp. 2481–2499, 2012.

[10] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[11] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: nonlinear phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.

[12] R. M. Willett and R. D. Nowak, "Platelets: a multiscale approach for recovering edges and surfaces in photon-limited medical imaging," *IEEE Transactions on Medical Imaging*, vol. 22, no. 3, pp. 332–350, 2003.

[13] A. Danielyan, V. Katkovnik, and K. Egiazarian, "BM3D frames and variational image deblurring," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1715–1728, 2012.

[14] W. Marais and R. Willett, "Proximal-gradient methods for Poisson image reconstruction with BM3D-based regularization," in *IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 2017, pp. 1–5.

[15] J. I. Tamir, F. Ong, S. Anand, E. Karasan, K. Wang, and M. Lustig, "Computational mri with physics-based constraints: Application to multicontrast and quantitative imaging," *IEEE Signal Processing Magazine*, vol. 37, no. 1, pp. 94–104, 2020.

[16] A. Mousavi, A. B. Patel, and R. G. Baraniuk, "A deep learning approach to structured signal recovery," in *2015 53rd annual allerton conference on communication, control, and computing (Allerton)*. IEEE, 2015, pp. 1336–1343.

[17] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, and A. Ashok, "Reconnet: Non-iterative reconstruction of images from compressively sensed measurements," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 449–458.

[18] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 2, pp. 295–307, 2016.

[19] J. Sun, H. Li, and Z. Xu, "Deep ADMM-Net for compressive sensing MRI," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2016, pp. 10–18.

[20] A. Mousavi and R. G. Baraniuk, "Learning to invert: Signal recovery via deep convolutional networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 2272–2276.

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

[22] J. R. Chang, C.-L. Li, B. Poczos, B. Vijaya Kumar, and A. C. Sankaranarayanan, "One network to solve them all–solving linear inverse problems using deep projection models," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5888–5897.

[23] A. Mousavi, G. Dasarathy, and R. G. Baraniuk, "DeepCodec: Adaptive sensing and recovery via deep convolutional neural networks," in *Allerton Conference on Communication, Control, and Computing*, 2017.

[24] K. Hammernik, T. Klatzer, E. Kobler, M. P. Recht, D. K. Sodickson, T. Pock, and F. Knoll, "Learning a variational network for reconstruction of accelerated mri data," *Magnetic resonance in medicine*, vol. 79, no. 6, pp. 3055–3071, 2018.

[25] A. Bora, A. Jalal, E. Price, and A. G. Dimakis, "Compressed sensing using generative models," in *International Conference on Machine Learning (ICML)*, 2017, pp. 537–546.

[26] A. Lucas, M. Iliadis, R. Molina, and A. K. Katsaggelos, "Using deep neural networks for inverse problems in imaging: beyond analytical methods," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 20–36, 2018.

[27] F. Knoll, K. Hammernik, C. Zhang, S. Moeller, T. Pock, D. K. Sodickson, and M. Akcakaya, "Deep-learning methods for parallel magnetic resonance imaging reconstruction: A survey of the current approaches, trends, and issues," *IEEE Signal Processing Magazine*, vol. 37, no. 1, pp. 128–140, 2020.

[28] https://www.gehealthcare.com/products/truefidelity, 2019.

[29] https://www.itnonline.com/content/fda-clears-ges-deep-learning-image-reconstruction-engine, 2019.

[30] C. Chen, Q. Chen, J. Xu, and V. Koltun, "Learning to see in the dark," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3291–3300.

[31] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 270–279.

[32] O. Liba, K. Murthy, Y.-T. Tsai, T. Brooks, T. Xue, N. Karnad, Q. He, J. T. Barron, D. Sharlet, R. Geiss, S. W. Hasinoff, and M. Levoy, "Handheld mobile photography in very low light," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, pp. 1–16, 2019.

[33] J. T. Barron and Y.-T. Tsai, "Fast fourier color constancy," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 886–894.

[34] U. S. Kamilov, I. N. Papadopoulos, M. H. Shoreh, A. Goy, C. Vonesch, M. Unser, and D. Psaltis, "Learning approach to optical tomography," *Optica*, vol. 2, no. 6, pp. 517–522, 2015.

[35] Y. Rivenson, Z. Göröcs, H. Günaydin, Y. Zhang, H. Wang, and A. Ozcan, "Deep learning microscopy," *Optica*, vol. 4, no. 11, pp. 1437–1443, 2017.

[36] E. Nehme, L. E. Weiss, T. Michaeli, and Y. Shechtman, "Deep-storm: super-resolution single-molecule microscopy by deep learning," *Optica*, vol. 5, no. 4, pp. 458–464, 2018.

[37] H. Pinkard, Z. Phillips, A. Babakhani, D. A. Fletcher, and L. Waller, "Deep learning for single-shot autofocus microscopy," *Optica*, vol. 6, no. 6, pp. 794–797, 2019.

[38] R. Horstmeyer, R. Y. Chen, B. Kappes, and B. Judkewitz, "Convolutional neural networks that teach microscopes how to image," *arXiv preprint arXiv:1709.07223*, 2017.

[39] E. Hershko, L. E. Weiss, T. Michaeli, and Y. Shechtman, "Multicolor localization microscopy and point-spread-function engineering by deep learning," *Optics express*, vol. 27, no. 5, pp. 6158–6183, 2019.

[40] M. Kellman, E. Bostan, M. Chen, and L. Waller, "Data-driven design for fourier ptychographic microscopy," in *2019 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 2019, pp. 1–8.

[41] M. Araya-Polo, J. Jennings, A. Adler, and T. Dahlke, "Deep-learning tomography," *The Leading Edge*, vol. 37, no. 1, pp. 58–66, 2018.

[42] T. M. Hansen and K. S. Cordua, "Efficient monte carlo sampling of inverse problems using a neural network-based forward—applied to gpr crosshole traveltime inversion," *Geophysical Journal International*, vol. 211, no. 3, pp. 1524–1533, 2017.

[43] L. Mosser, O. Dubrule, and M. J. Blunt, "Stochastic seismic waveform inversion using generative adversarial networks as a geological prior," *Mathematical Geosciences*, vol. 52, no. 1, pp. 53–79, 2020.

[44] L. Yang, D. Zhang, and G. E. Karniadakis, "Physics-informed generative adversarial networks for stochastic differential equations," *SIAM Journal on Scientific Computing*, vol. 42, no. 1, pp. A292–A317, 2020.

[45] A. Sinha, J. Lee, S. Li, and G. Barbastathis, "Lensless computational imaging through deep learning," *Optica*, vol. 4, no. 9, pp. 1117–1125, 2017.

[46] Y. Rivenson, Y. Zhang, H. Günaydın, D. Teng, and A. Ozcan, "Phase recovery and holographic image reconstruction using deep learning in neural networks," *Light: Science & Applications*, vol. 7, no. 2, pp. 17 141–17 141, 2018.

[47] Y. Wu, Y. Rivenson, Y. Zhang, Z. Wei, H. Günaydin, X. Lin, and A. Ozcan, "Extended depth-of-field in holographic imaging using deep-learning-based autofocusing and phase recovery," *Optica*, vol. 5, no. 6, pp. 704–710, 2018.

[48] M. Lyu, W. Wang, H. Wang, H. Wang, G. Li, N. Chen, and G. Situ, "Deep-learning-based ghost imaging," *Scientific reports*, vol. 7, no. 1, pp. 1–6, 2017.

[49] N. Borhani, E. Kakkava, C. Moser, and D. Psaltis, "Learning to see through multimode fibers," *Optica*, vol. 5, no. 8, pp. 960–966, 2018.

[50] Y. Li, Y. Xue, and L. Tian, "Deep speckle correlation: a deep learning approach toward scalable imaging through scattering media," *Optica*, vol. 5, no. 10, pp. 1181–1190, 2018.

[51] Y. Sun, Z. Xia, and U. S. Kamilov, "Efficient and accurate inversion of multiple scattering with deep learning," *Optics express*, vol. 26, no. 11, pp. 14 678–14 688, 2018.

[52] C. A. Metzler, F. Heide, P. Rangarajan, M. M. Balaji, A. Viswanath, A. Veeraraghavan, and R. G. Baraniuk, "Deep-inverse correlography: towards real-time high-resolution non-line-of-sight imaging," *Optica*, vol. 7, no. 1, pp. 63–71, 2020.

[53] G. Barbastathis, A. Ozcan, and G. Situ, "On the use of deep learning for computational imaging," *Optica*, vol. 6, no. 8, pp. 921–943, 2019.

[54] A. N. Tikhonov, "On the stability of inverse problems," in *Dokl. Akad. Nauk SSSR*, vol. 39, 1943, pp. 195–198.

[55] S. J. Wright, R. D. Nowak, and M. A. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2479–2493, 2009.

[56] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 948–958, 2010.

[57] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM Journal on Imaging Sciences*, vol. 1, no. 3, pp. 248–272, 2008.

[58] R. C. Gonzalez and R. E. Woods, "Image processing," *Digital image processing*, vol. 2, p. 1, 2007.

[59] E. Caroli, J. Stephen, G. Di Cocco, L. Natalucci, and A. Spizzichino, "Coded aperture imaging in x-and gamma-ray astronomy," *Space Science Reviews*, vol. 45, no. 3-4, pp. 349–403, 1987.

[60] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE transactions on image processing*, vol. 13, no. 10, pp. 1327–1344, 2004.

[61] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE transactions on image processing*, vol. 19, no. 11, pp. 2861–2873, 2010.

[62] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 2000, pp. 417–424.

[63] E. J. Candes, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on pure and applied mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.

[64] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE signal processing magazine*, vol. 25, no. 2, pp. 83–91, 2008.

[65] K. P. Pruessmann, M. Weiger, M. B. Scheidegger, and P. Boesiger, "SENSE: sensitivity encoding for fast MRI," *Magnetic Resonance in Medicine*, vol. 42, no. 5, pp. 952–962, 1999.

[66] G. Beylkin, "Discrete radon transform," *IEEE transactions on acoustics, speech, and signal processing*, vol. 35, no. 2, pp. 162–172, 1987.

[67] C. Fienup and J. Dainty, "Phase retrieval and image reconstruction for astronomy," *Image recovery: theory and application*, vol. 231, p. 275, 1987.

[68] J. Bertolotti, E. G. Van Putten, C. Blum, A. Lagendijk, W. L. Vos, and A. P. Mosk, "Non-invasive imaging through opaque scattering layers," *Nature*, vol. 491, no. 7423, pp. 232–234, 2012.

[69] L. Tian, X. Li, K. Ramchandran, and L. Waller, "Multiplexed coded illumination for fourier ptychography with an led array microscope," *Biomedical optics express*, vol. 5, no. 7, pp. 2376–2389, 2014.

[70] C. A. Metzler, M. K. Sharma, S. Nagesh, R. G. Baraniuk, O. Cossairt, and A. Veeraraghavan, "Coherent inverse scattering via transmission matrices: Efficient phase retrieval algorithms and a public dataset," in *2017 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 2017, pp. 1–16.

[71] G. K. Wallace, "The jpeg still picture compression standard," *IEEE transactions on consumer electronics*, vol. 38, no. 1, pp. xviii–xxxiv, 1992.

[72] M. W. Marcellin, M. J. Gormish, A. Bilgin, and M. P. Boliek, "An overview of jpeg-2000," in *Data Compression Conference, 2000. Proceedings. DCC 2000*. IEEE, 2000, pp. 523–541.

[73] B. S. Manjunath, P. Salembier, and T. Sikora, *Introduction to MPEG-7: multimedia content description interface*. John Wiley & Sons, 2002, vol. 1.

[74] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[75] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[76] A. Bora, E. Price, and A. G. Dimakis, "Ambientgan: Generative models from lossy measurements." *ICLR*, vol. 2, p. 5, 2018.

[77] A. Kamath, S. Karmalkar, and E. Price, "Lower bounds for compressed sensing with generative models," *arXiv preprint arXiv:1912.02938*, 2019.

[78] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4509–4522, 2017.

[79] J. C. Ye, Y. Han, and E. Cha, "Deep convolutional framelets: A general deep learning framework for inverse problems," *SIAM Journal on Imaging Sciences*, vol. 11, no. 2, pp. 991–1048, 2018.

[80] S. Diamond, V. Sitzmann, F. Heide, and G. Wetzstein, "Unrolled optimization with deep priors," *arXiv preprint arXiv:1705.08041*, 2017.

[81] T. Meinhardt, M. Moeller, C. Hazirbas, and D. Cremers, "Learning proximal operators: Using denoising networks for regularizing inverse imaging problems," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1799–1808.

[82] H. K. Aggarwal, M. P. Mani, and M. Jacob, "MoDL: Model based deep learning architecture for inverse problems," *IEEE Transactions on Medical Imaging*, 2018.

[83] J. Adler and O. Öktem, "Learned primal-dual reconstruction," *IEEE Transactions on Medical Imaging*, vol. 37, no. 6, pp. 1322–1332, 2018.

[84] D. Gilton, G. Ongie, and R. Willett, "Neumann networks for linear inverse problems in imaging," *IEEE Transactions on Computational Imaging*, 2019.

[85] C. A. Metzler, A. Mousavi, R. Heckel, and R. G. Baraniuk, "Unsupervised learning with Stein's unbiased risk estimator," *arXiv preprint arXiv:1805.10531*, 2018.

[86] M. Zhussip, S. Soltanayev, and S. Y. Chun, "Training deep learning based image denoisers from undersampled measurements without ground truth and without image prior," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 255–10 264.

[87] J. I. Tamir, X. Y. Stella, and M. Lustig, "Unsupervised deep basis pursuit: Learning reconstruction without ground-truth data," in *Proceedings of the 27th Annual Meeting of ISMRM*, 2019.

[88] C. Metzler, A. Mousavi, and R. G. Baraniuk, "Learned d-amp: Principled neural network based compressive image recovery," in *Advances in Neural Information Processing Systems*, 2017, pp. 1772–1783.

[89] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, "Plug-and-play priors for model based reconstruction," in *2013 IEEE Global Conference on Signal and Information Processing*. IEEE, 2013, pp. 945–948.

[90] Y. Romano, M. Elad, and P. Milanfar, "The little engine that could: Regularization by denoising (red)," *SIAM Journal on Imaging Sciences*, vol. 10, no. 4, pp. 1804–1844, 2017.

[91] D. Engin, A. Genç, and H. Kemal Ekenel, "Cycle-dehaze: Enhanced cyclegan for single image dehazing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 825–833.

[92] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, "Deblurgan: Blind motion deblurring using conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8183–8192.

[93] M. Asim, F. Shamshad, and A. Ahmed, "Blind image deconvolution using deep generative priors," *arXiv preprint arXiv:1802.04073*, 2018.

[94] R. Anirudh, J. J. Thiagarajan, B. Kailkhura, and T. Bremer, "An unsupervised approach to solving inverse problems using generative adversarial networks," *arXiv preprint arXiv:1805.07281*, 2018.

[95] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, "Noise2noise: Learning image restoration without clean data," *arXiv preprint arXiv:1803.04189*, 2018.

[96] A. Pajot, E. de Bezenac, and P. Gallinari, "Unsupervised adversarial image reconstruction," 2018.

[97] B. Zhu, J. Z. Liu, S. F. Cauley, B. R. Rosen, and M. S. Rosen, "Image reconstruction by domain-transform manifold learning," *Nature*, vol. 555, no. 7697, pp. 487–492, 2018.

[98] M. T. McCann, K. H. Jin, and M. Unser, "Convolutional neural networks for inverse problems in imaging: A review," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 85–95, 2017.

[99] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Springer, 2015, pp. 234–241.

[100] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-point algorithms for inverse problems in science and engineering*. Springer, 2011, pp. 185–212.

[101] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *International Conference on Machine Learning (ICML)*, 2010, pp. 399–406.

[102] Y. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1256–1272, 2017.

[103] M. Mardani, Q. Sun, S. Vasawanala, V. Papyan, H. Monajemi, J. Pauly, and D. Donoho, "Neural proximal gradient descent for compressive imaging," *arXiv preprint arXiv:1806.03963*, 2018.

[104] R. Liu, S. Cheng, L. Ma, X. Fan, and Z. Luo, "Deep proximal unrolling: Algorithmic framework, convergence analysis and applications," *IEEE Transactions on Image Processing*, vol. 28, no. 10, pp. 5013–5026, 2019.

[105] U. Schmidt and S. Roth, "Shrinkage fields for effective image restoration," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 2774–2781.

[106] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2017.

[107] S. Ravishankar, I. Y. Chun, and J. A. Fessier, "Physics-driven deep training of dictionary-based algorithms for MR image reconstruction," in *Asilomar Conference on Signals, Systems, and Computers*, 2017, pp. 1859–1863.

[108] S. Ravishankar, A. Lahiri, C. Blocker, and J. A. Fessler, "Deep dictionary-transform learning for image reconstruction," in *IEEE International Symposium on Biomedical Imaging (ISBI)*, 2018, pp. 1208–1212.

[109] Y. Chun and J. A. Fessler, "Deep BCD-net using identical encoding-decoding CNN structures for iterative image recovery," in *IEEE Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*. IEEE, 2018, pp. 1–5.

[110] M. Borgerding, P. Schniter, and S. Rangan, "Amp-inspired deep networks for sparse linear inverse problems," *IEEE Transactions on Signal Processing*, vol. 65, no. 16, pp. 4293–4308, 2017.

[111] B. Yaman, S. A. H. Hosseini, S. Moeller, J. Ellermann, K. Uğurbil, and M. Akçakaya, "Self-supervised learning of physics-based reconstruction neural networks without fully-sampled reference data," *arXiv preprint arXiv:1912.07669*, 2019.

[112] C. M. Stein, "Estimation of the mean of a multivariate normal distribution," *The annals of Statistics*, pp. 1135–1151, 1981.

[113] Y. C. Eldar, "Generalized sure for exponential families: Applications to regularization," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 471–481, 2009.

[114] S. Soltanayev and S. Y. Chun, "Training deep learning based denoisers without ground truth data," in *Advances in Neural Information Processing Systems*, 2018, pp. 3257–3267.

[115] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Transactions on image processing*, vol. 16, no. 8, pp. 2080–2095, 2007.

[116] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[117] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends® in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[118] H. Gupta, K. H. Jin, H. Q. Nguyen, M. T. McCann, and M. Unser, "CNN-based projected gradient descent for consistent CT image reconstruction," *IEEE Transactions on Medical Imaging*, vol. 37, no. 6, pp. 1440–1453, 2018.

[119] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proceedings of the National Academy of Sciences*, vol. 106, no. 45, pp. 18 914–18 919, 2009.

[120] Q. Lei, A. Jalal, I. S. Dhillon, and A. G. Dimakis, "Inverting deep generative models, one layer at a time," in *Advances in Neural Information Processing Systems*, 2019, pp. 13 910–13 919.

[121] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[122] C. Hegde, M. Wakin, and R. G. Baraniuk, "Random projections for manifold learning," in *Advances in neural information processing systems*, 2008, pp. 641–648.

[123] R. G. Baraniuk and M. B. Wakin, "Random projections of smooth manifolds," *Foundations of computational mathematics*, vol. 9, no. 1, pp. 51–77, 2009.

[124] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Transactions on Information Theory*, vol. 56, no. 4, pp. 1982–2001, 2010.

[125] Y. C. Eldar and M. Mishali, "Robust recovery of signals from a structured union of subspaces," *IEEE Transactions on Information Theory*, vol. 55, no. 11, pp. 5302–5316, 2009.

[126] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[127] D. L. Donoho, "Compressed sensing," *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[128] P. J. Bickel, Y. Ritov, A. B. Tsybakov *et al.*, "Simultaneous analysis of lasso and dantzig selector," *The Annals of Statistics*, vol. 37, no. 4, pp. 1705–1732, 2009.

[129] E. J. Candes, "The restricted isometry property and its implications for compressed sensing," *Comptes rendus mathematique*, vol. 346, no. 9-10, pp. 589–592, 2008.

[130] Z. Liu and J. Scarlett, "Sample complexity lower bounds for compressive sensing with generative models," in *NeurIPS 2019 Workshop on Solving Inverse Problems with Deep Networks*, 2019.

[131] A. Jalal, L. Liu, A. G. Dimakis, and C. Caramanis, "Robust compressed sensing of generative models," *arXiv preprint*, 2020.

[132] S. Jalali and X. Yuan, "Solving linear inverse problems using generative models," in *2019 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2019, pp. 512–516.

[133] B. Aubin, B. Loureiro, A. Baker, F. Krzakala, and L. Zdeborová, "Exact asymptotics for phase retrieval and compressed sensing with random generative priors," *arXiv preprint arXiv:1912.02008*, 2019.

[134] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," *arXiv preprint arXiv:1809.11096*, 2018.

[135] G. Daras, A. Odena, H. Zhang, and A. G. Dimakis, "Your local gan: Designing two dimensional local attention mechanisms for generative models," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[136] P. Hand and V. Voroninski, "Global guarantees for enforcing deep generative priors by empirical risk," *arXiv preprint arXiv:1705.07576*, 2017.

[137] V. Shah and C. Hegde, "Solving linear inverse problems using gan priors: An algorithm with provable guarantees," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4609–4613.

[138] F. L. Gómez, A. Eftekhari, and V. Cevher, "Fast and provable admm for learning with generative priors," in *Advances in Neural Information Processing Systems*, 2019, pp. 12 004–12 016.

[139] M. Mardani, E. Gong, J. Y. Cheng, S. Vasanawala, G. Zaharchuk, M. Alley, N. Thakur, S. Han, W. Dally, J. M. Pauly, and L. Xing, "Deep generative adversarial networks for compressed sensing automates mri," *arXiv preprint arXiv:1706.00051*, 2017.

[140] S. Tripathi, Z. C. Lipton, and T. Q. Nguyen, "Correction by projection: Denoising images with generative adversarial networks," *arXiv preprint arXiv:1803.04477*, 2018.

[141] R. Anirudh, J. J. Thiagarajan, B. Kailkhura, and T. Bremer, "Mimicgan: Robust projection onto image manifolds with corruption mimicking," *arXiv preprint arXiv:1912.07748*, 2019.

[142] P. Hand, O. Leong, and V. Voroninski, "Phase retrieval under a generative prior," in *Advances in Neural Information Processing Systems*, 2018, pp. 9136–9146.

[143] M. Asim, F. Shamshad, and A. Ahmed, "Solving bilinear inverse problems using deep generative priors," *CoRR, abs/1802.04073*, vol. 3, no. 4, p. 8, 2018.

[144] S. Qiu, X. Wei, and Z. Yang, "Robust one-bit recovery via relu generative networks: Improved statistical rates and global landscape analysis," *arXiv preprint arXiv:1908.05368*, 2019.

[145] M. Dhar, A. Grover, and S. Ermon, "Modeling sparse deviations for compressed sensing using generative models," *arXiv preprint arXiv:1807.01442*, 2018.

[146] M. Kabkab, P. Samangouei, and R. Chellappa, "Task-aware compressed sensing with generative adversarial networks," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[147] A. K. Fletcher, S. Rangan, and P. Schniter, "Inference in deep networks in high dimensions," in *2018 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2018, pp. 1884–1888.

[148] A. K. Fletcher, P. Pandit, S. Rangan, S. Sarkar, and P. Schniter, "Plug-in estimation in high-dimensional linear inverse problems: A rigorous analysis," in *Advances in Neural Information Processing Systems*, 2018, pp. 7440–7449.

[149] G. Song, Z. Fan, and J. Lafferty, "Surfing: Iterative optimization over incrementally trained deep networks," in *Advances in Neural Information Processing Systems*, 2019, pp. 15 008–15 017.

[150] E. M. Lindgren, J. Whang, and A. G. Dimakis, "Conditional sampling from invertible generative models with applications to inverse problems," *arXiv preprint arXiv:2002.11743*, 2020.

[151] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[152] K. Armanious, C. Jiang, S. Abdulatif, T. Küstner, S. Gatidis, and B. Yang, "Unsupervised medical image translation using Cycle-MedGAN," in *European Signal Processing Conference (EUSIPCO)*, 2019, pp. 1–5.

[153] T. M. Quan, T. Nguyen-Duc, and W.-K. Jeong, "Compressed sensing MRI reconstruction using a generative adversarial network with a cyclic loss," *IEEE Transactions on Medical Imaging*, vol. 37, no. 6, pp. 1488–1497, 2018.

[154] P. Hand and B. Joshi, "Global guarantees for blind demodulation with generative priors," *arXiv preprint arXiv:1905.12576*, 2019.

[155] C. K. Sønderby, J. Caballero, L. Theis, W. Shi, and F. Huszár, "Amortised map inference for image super-resolution," *arXiv preprint arXiv:1610.04490*, 2016.

[156] R. A. DeVore and G. G. Lorentz, *Constructive approximation*. Springer Science & Business Media, 1993, vol. 303.

[157] D. L. Donoho, I. M. Johnstone, G. Kerkyacharian, and D. Picard, "Density estimation by wavelet thresholding," *The Annals of Statistics*, pp. 508–539, 1996.

[158] B. Delyon and A. Juditsky, "On minimax wavelet estimators," *Applied and Computational Harmonic Analysis*, vol. 3, no. 3, pp. 215–228, 1996.

[159] J. Lafferty, H. Liu, and L. Wasserman, "Minimax theory," http://www.stat.cmu.edu/~larry/=sml/Minimax.pdf, 2008, [Online; accessed 07-January-2019].

[160] S. Efromovich, "Conditional density estimation in a regression setting," *The Annals of Statistics*, vol. 35, no. 6, pp. 2504–2535, 2007.

[161] K. Bertin, C. Lacour, and V. Rivoirard, "Adaptive pointwise estimation of conditional density function," in *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques*, vol. 52, no. 2. Institut Henri Poincaré, 2016, pp. 939–980.

[162] A. Wirgin, "The inverse crime," *arXiv preprint math-ph/0401050*, 2004.

[163] M. Guerquin-Kern, L. Lejeune, K. P. Pruessmann, and M. Unser, "Realistic analytical phantoms for parallel magnetic resonance imaging," *IEEE Transactions on Medical Imaging*, vol. 31, no. 3, pp. 626–636, 2011.

[164] X. Mao, C. Shen, and Y.-B. Yang, "Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2016, pp. 2802–2810.

[165] V. Antun, F. Renna, C. Poon, B. Adcock, and A. C. Hansen, "On instabilities of deep learning in image reconstruction-does ai come at a cost?" *arXiv preprint arXiv:1902.05300*, 2019.

[166] N. M. Gottschling, V. Antun, B. Adcock, and A. C. Hansen, "The troublesome kernel: why deep learning for inverse problems is typically unstable," *arXiv preprint arXiv:2001.01258*, 2020.

[167] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9446–9454.

[168] R. Heckel and P. Hand, "Deep decoder: Concise image representations from untrained non-convolutional networks," *arXiv preprint arXiv:1810.03982*, 2018.

[169] D. Van Veen, A. Jalal, E. Price, S. Vishwanath, and A. G. Dimakis, "Compressed sensing with deep image prior and learned regularization," *arXiv preprint arXiv:1806.06438*, 2018.

[170] R. Heckel and M. Soltanolkotabi, "Denoising and regularization via exploiting the structural bias of convolutional generators," *arXiv preprint arXiv:1910.14634*, 2019.

[171] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse mri: The application of compressed sensing for rapid mr imaging," *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, vol. 58, no. 6, pp. 1182–1195, 2007.

[172] M. R. Kellman, E. Bostan, N. A. Repina, and L. Waller, "Physics-based learned design: optimized coded-illumination for quantitative phase imaging," *IEEE Transactions on Computational Imaging*, vol. 5, no. 3, pp. 344–353, 2019.

[173] A. Chakrabarti, "Learning sensor multiplexing design through back-propagation," in *Advances in Neural Information Processing Systems*, 2016, pp. 3081–3089.

[174] V. Sitzmann, S. Diamond, Y. Peng, X. Dun, S. Boyd, W. Heidrich, F. Heide, and G. Wetzstein, "End-to-end optimization of optics and image processing for achromatic extended depth of field and super-resolution imaging," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–13, 2018.

[175] J. Chang and G. Wetzstein, "Deep optics for monocular depth estimation and 3d object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 10 193–10 202.

[176] Y. Wu, V. Boominathan, H. Chen, A. Sankaranarayanan, and A. Veeraraghavan, "Phasecam3d—learning phase masks for passive single view depth estimation," in *2019 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 2019, pp. 1–12.

[177] C. A. Metzler, H. Ikoma, Y. Peng, and G. Wetzstein, "Deep optics for single-shot high-dynamic-range imaging," *arXiv preprint arXiv:1908.00620*, 2019.

[178] Y. Peng, Q. Sun, X. Dun, G. Wetzstein, W. Heidrich, and F. Heide, "Learned large field-of-view imaging with thin-plate optics," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, p. 219, 2019.

[179] G. Jagatap and C. Hegde, "Algorithmic guarantees for inverse imaging with untrained network priors," *arXiv preprint arXiv:1906.08763*, 2019.

[180] ——, "Phase retrieval using untrained neural network priors," 2019.

[181] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, "Self-taught learning: transfer learning from unlabeled data," in *Proceedings of the 24th international conference on Machine learning*, 2007, pp. 759–766.

[182] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual domain adaptation: A survey of recent advances," *IEEE signal processing magazine*, vol. 32, no. 3, pp. 53–69, 2015.

[183] C. A. Metzler, P. Schniter, A. Veeraraghavan, and R. G. Baraniuk, "prDeep: Robust phase retrieval with a flexible deep network," in *International Conference on Machine Learning*, 2018, pp. 3501–3510.

[184] C. A. Metzler and G. Wetzstein, "Deep S^3PR: Simultaneous source separation and phase retrieval using deep generative models," *arXiv preprint arXiv:2002.05856*, 2020.

[185] M. Deng, S. Li, A. Goy, I. Kang, and G. Barbastathis, "Learning to synthesize: Robust phase retrieval at low photon counts," *Light: Science & Applications*, vol. 9, no. 1, pp. 1–16, 2020.

[186] J. Adler and O. Öktem, "Deep Bayesian inversion," *arXiv preprint arXiv:1811.05910*, 2018.

[187] J. Schlemper, D. C. Castro, W. Bai, C. Qin, O. Oktay, J. Duan, A. N. Price, J. Hajnal, and D. Rueckert, "Bayesian deep learning for accelerated MR image reconstruction," in *International Workshop on Machine Learning for Medical Image Reconstruction*. Springer, 2018, pp. 64–71.