



HAL
open science

Training Adaptive Reconstruction Networks for Blind Inverse Problems

Alban Gossard, Pierre Weiss

► **To cite this version:**

Alban Gossard, Pierre Weiss. Training Adaptive Reconstruction Networks for Blind Inverse Problems. 2022. hal-03585120v3

HAL Id: hal-03585120

<https://hal.science/hal-03585120v3>

Preprint submitted on 13 Nov 2022 (v3), last revised 13 Dec 2023 (v5)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Training Adaptive Reconstruction Networks for Blind Inverse Problems

Alban Gossard^{1,2}

alban.paul.gossard@gmail.com

Pierre Weiss^{1,2}

pierre.armand.weiss@gmail.com

¹ Institut de Mathématiques de Toulouse; UMR5219; Université de Toulouse; CNRS

² UPS, F-31062 Toulouse Cedex 9, France

November 13, 2022

Abstract

Neural networks have recently allowed solving many ill-posed inverse problems with unprecedented performance. Physics informed approaches already progressively replace carefully hand-crafted reconstruction algorithms in real applications. However, these networks suffer from a major defect: when trained on a given forward operator, they do not generalize well to a different one. The aim of this paper is twofold. First, we show through various applications that training the network with a family of forward operators allows solving the adaptivity problem without compromising the reconstruction quality significantly. Second, we illustrate that this training procedure allows tackling challenging blind inverse problems. Our experiments include partial Fourier sampling problems arising in magnetic resonance imaging (MRI), computerized tomography (CT) and image deblurring.

Index terms— Blind inverse problems, self-calibration, adaptivity, model based reconstruction, convolutional neural network, unrolled networks, MRI reconstruction, computerized tomography, blind deblurring

1 Introduction

The primary contribution of this paper is the design of model-based neural networks to solve *families* of blind linear inverse problems. Many sensing devices like cameras, magnetic resonance imaging (MRI) or computerized tomography (CT) systems measure a signal $\mathbf{x} \in \mathbb{C}^N$ through a linear operator $\mathbf{A}(\boldsymbol{\theta}) \in \mathbb{C}^{M \times N}$. The parameter $\boldsymbol{\theta} \in \mathbb{R}^P$ characterizes the sensing operator. For instance, it can encode the point spread function in image deblurring, the projection angles in CT or the Fourier sampling locations in MRI. This leads to measurements of the form:

$$\mathbf{y} = \mathcal{P}(\mathbf{A}(\boldsymbol{\theta})\mathbf{x}), \quad (1)$$

where $\mathcal{P} : \mathbb{C}^M \rightarrow \mathbb{C}^M$ is a perturbation (e.g. additive Gaussian noise, quantization). A model based inverse problem consists in recovering an estimate $\hat{\mathbf{x}}$ of \mathbf{x} from \mathbf{y} and $\mathbf{A}(\boldsymbol{\theta})$. If the parameter $\boldsymbol{\theta}$ is unknown, then we speak of a blind inverse problem.

In this paper, we focus on neural network based reconstruction. We consider mappings of the form:

$$\begin{aligned} \mathcal{N} : \mathbb{R}^D \times \mathbb{R}^P \times \mathbb{C}^M &\rightarrow \mathbb{R}^N \\ (\mathbf{w}, \boldsymbol{\theta}, \mathbf{y}) &\mapsto \mathcal{N}[\mathbf{w}, \boldsymbol{\theta}, \mathbf{y}]. \end{aligned} \quad (2)$$

Given a weight $\mathbf{w} \in \mathbb{R}^D$, a forward operator parametrization $\boldsymbol{\theta}$ and a measurement vector \mathbf{y} , the network \mathcal{N} outputs an estimate $\hat{\mathbf{x}} = \mathcal{N}[\mathbf{w}, \boldsymbol{\theta}, \mathbf{y}]$. Given a forward operator $\mathbf{A}(\boldsymbol{\theta}_0)$, the traditional procedure to optimize the weights \mathbf{w} , is to minimize the following empirical risk:

$$\inf_{\mathbf{w} \in \mathbb{R}^D} \frac{1}{2I} \sum_{i=1}^I \|\mathcal{N}[\mathbf{w}, \boldsymbol{\theta}_0, \mathbf{y}_i] - \mathbf{x}_i\|_2^2, \quad (3)$$

where $(\mathbf{x}_i)_{1 \leq i \leq I}$ is a collection of training images and (\mathbf{y}_i) is the corresponding collection of measurements generated using (1). That is, we wish the reconstruction mapping to output images close in average to

the true underlying signals. In this paper, we explore a seemingly minor variation of this principle by solving:

$$\inf_{\mathbf{w} \in \mathbb{R}^D} E(\mathbf{w}) \stackrel{\text{def}}{=} \mathbb{E} \left[\frac{1}{2I} \sum_{i=1}^I \|\mathcal{N}[\mathbf{w}, \boldsymbol{\theta}, \mathbf{y}_i] - \mathbf{x}_i\|_2^2 \right], \quad (4)$$

where the expectation is taken with respect to the parameter $\boldsymbol{\theta}$ considered as a random variable. That is, we train our reconstruction mapping on a *family of operators*. The main motivation for this modification is twofold. First, we want to address a lack of adaptivity for the standard training procedure. Second, we want to use the resulting reconstruction mapping to solve blind inverse problems. Let us discuss these two points in more depth.

The adaptivity issue While model-based reconstruction networks provide state-of-the-art results in a large panel of applications, it is now well established that they suffer from a *lack of adaptivity*. This means that a network trained for a specific operator $\mathbf{A}(\boldsymbol{\theta}_0)$ may have a significant performance drop if used for another operator $\mathbf{A}(\boldsymbol{\theta}_1)$. This drop can be evaluated as follows. Let $\boldsymbol{\theta}_0 \neq \boldsymbol{\theta}_1$ denote two different operator parametrizations. Let $\mathbf{y}_0 = \mathcal{P}(\mathbf{A}(\boldsymbol{\theta}_0)\mathbf{x})$ and $\mathbf{y}_1 = \mathcal{P}(\mathbf{A}(\boldsymbol{\theta}_1)\mathbf{x})$. Assume that \mathbf{w}_0^* and \mathbf{w}_1^* are the weights of a reconstruction network optimized for $\mathbf{A}(\boldsymbol{\theta}_0)$ and $\mathbf{A}(\boldsymbol{\theta}_1)$ respectively. We compare the quality of $\mathcal{N}(\mathbf{w}_1^*, \boldsymbol{\theta}_0, \mathbf{y}_0)$ and $\mathcal{N}(\mathbf{w}_0^*, \boldsymbol{\theta}_0, \mathbf{y}_0)$ in the second and third rows of Fig. 1. Observe the significant performance difference.

To avoid this pitfall, we propose to train the network by minimizing (4). We will carefully evaluate the performance of the resulting networks in Section 6 for MR image reconstruction from under-sampled data, CT imaging with limited angles and image deblurring. We conclude that this learning approach yields a reconstruction network which is significantly more stable to variations of the forward operator. In addition, the performance of an unrolled network trained on a restricted family is only marginally worse than that of a network that would be trained and used for a single operator. It therefore provides a satisfactory answer to the adaptivity issue. We also address several questions raised by our methodology. Can the unrolled network trained on a family extrapolate to unseen operators? How to sample the space of admissible operators \mathcal{A} ? What is the gain of our approach in comparison to more “universal approaches” such as plug&play priors?

Blind inverse problems Assume that we observe $\mathbf{y}_1 = \mathcal{P}(\mathbf{A}(\boldsymbol{\theta}_1)\mathbf{x})$. Unfortunately, we only have access to an approximate knowledge $\mathbf{A}(\boldsymbol{\theta}_0)$ of the forward model. This can be due to an imprecise calibration of the sensing device or to the motion of a patient in a scanner for instance. A problem solved with a model mismatch (i.e. with the operator $\mathbf{A}(\boldsymbol{\theta}_0)$ in place of $\mathbf{A}(\boldsymbol{\theta}_1)$), can lead to catastrophic reconstruction results, as illustrated in the last row of Fig. 1.

The second contribution of this work is to propose a systematic approach to recover an estimate $\hat{\boldsymbol{\theta}}_1$ of $\boldsymbol{\theta}_1$ from the observation \mathbf{y}_1 . We show that unrolled networks trained on a family of forward models provide a powerful tool to solve several blind inverse problems. The idea is simply to minimize the data consistency error

$$\hat{\boldsymbol{\theta}} \in \arg \min_{\boldsymbol{\theta} \in \Theta} \frac{1}{2} \|\mathbf{A}(\boldsymbol{\theta})\mathcal{N}[\mathbf{w}, \boldsymbol{\theta}, \mathbf{y}] - \mathbf{y}\|_2^2. \quad (5)$$

The reconstructed image $\hat{\mathbf{x}} \stackrel{\text{def}}{=} \mathcal{N}[\mathbf{w}, \hat{\boldsymbol{\theta}}, \mathbf{y}]$ is defined as the output of the unrolled neural network.

This consistency principle is spread massively in the literature of blind inverse problems. The main contribution here is to plug it with a specific training procedure on a family of forward operators.

2 Related works

Let us contextualize this work.

Regularization theory From a historical perspective, the first inverse problem solvers were based on simple inverses or approximate inverses of $\mathbf{A}(\boldsymbol{\theta})$. This approach provides low quality results when the matrix $\mathbf{A}(\boldsymbol{\theta})$ has a non trivial kernel or when the conditioning number of $\mathbf{A}^*(\boldsymbol{\theta})$ is high. In those cases, it is critical to use regularization terms. For long (~ 1960 -2000), simple quadratic terms (Tikhonov) dominated the scientific landscape. Around 1990, a second research trend appeared with convex, nonlinear regularizers such as total variation [61]. This area culminated with the development of the compressed

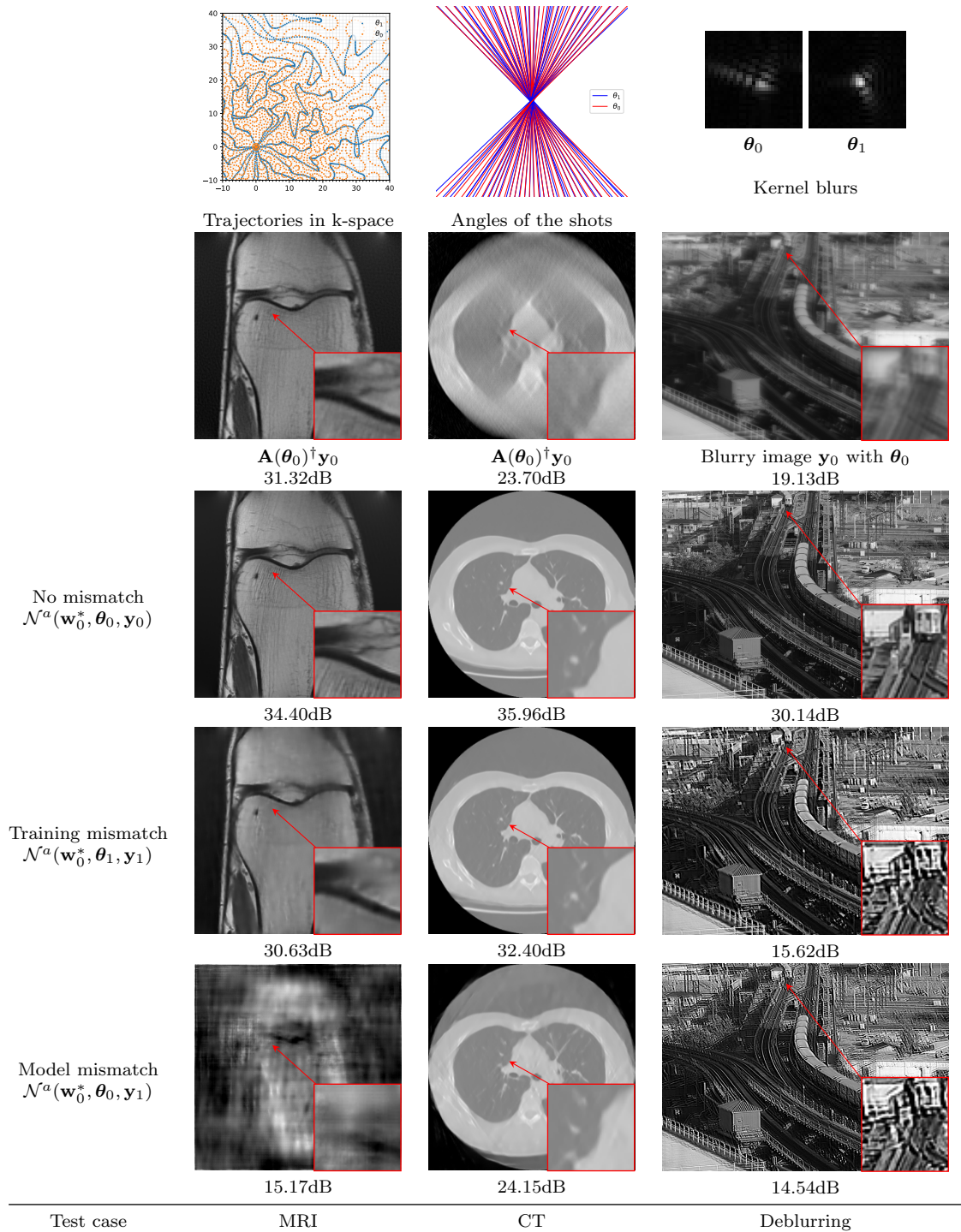


Figure 1: Examples of the issues addressed in this paper. *1st row*: description of the forward operators parameterized by θ_0 and θ_1 . *2nd row*: pseudo-inverse reconstruction of $\mathbf{y}_0 = \mathbf{A}(\theta_0)\mathbf{x}$. *3rd row*: reconstruction with no model or training mismatch. *4th row*: reconstruction with a training mismatch. *Last row*: reconstruction with a model mismatch (blind). All the models are an unrolled ADMM trained on $\mathbf{A}(\theta_0)$. The reconstruction PSNR is provided below each image.

sensing theory [15, 50]. Starting from 2015, impressive performance gains have occurred with the advent of neural networks. They seem to be likely to replace the initial methods in a growing number of technologies [72].

Learned reconstruction There are two main approaches to attack reconstruction problems using machine learning [7]. A first solution is *end-to-end networks* where the neural network is agnostic to the operator $\mathbf{A}(\boldsymbol{\theta})$. It gets trained through pairs $(\mathbf{y}_i, \mathbf{x}_i)$ generated with the model (1). A popular example is AUTOMAP [82]. In this technology, the network needs to infer the forward model from the training data. This usually requires a huge amount of training data for large M and N .

The other possibility is *model-based* reconstruction networks that are defined as mappings of the form (2). They are often praised for the fact that they require less training data and benefit from a higher interpretability. Two popular approaches among this class are:

- *Denoising nets*: There, the reconstruction network performs a rough inversion followed by a denoising network such as a U-Net, to remove the remaining artifacts, see e.g. [38].
- *Unrolled nets*: Many efficient iterative methods have been developed to solve convex optimization problems (proximal gradient descent, Douglas-Rachford, ADMM, Primal-Dual, ...) [21]. They have the general form:

$$\mathbf{x}_{k+1} = \text{prox}_R(M(\mathbf{A}(\boldsymbol{\theta}), \mathbf{y}, \mathbf{x}_k)), \quad (6)$$

for $k = 1$ to $K \in \mathbb{N}$. The mapping M is linear and can be interpreted as a crude way to invert the operator, in the sense that $\mathbf{A}(\boldsymbol{\theta})M(\mathbf{A}(\boldsymbol{\theta}), \mathbf{y}, \mathbf{x}_k) \simeq \mathbf{y}$. The term prox_R can be interpreted as a way to regularize (denoise) the remaining artifacts. The so-called plug&play priors [71] fit in this category.

The unrolled networks draw their inspiration from (6). They consist in replacing the handcrafted or learned proximal operator prox_R by a sequence of neural networks $(\mathcal{N}_k[\mathbf{w}_k])_{1 \leq k \leq K}$ promoting an output \mathbf{x}_K similar to the training images. The difference with the plug&play priors is that the weights \mathbf{w}_k are trained specifically for a given operator \mathbf{A} . Examples of approaches in this category include [68, 23, 1, 79, 25, 2, 3, 37, 46]. These algorithms are currently among the most efficient for MRI reconstruction [52].

For completeness, let us mention that a popular alternative consists in synthesizing the images \mathbf{x} with generative models [10, 8]. Compared to the approaches mentioned above, it typically suffers from a higher computational cost. Indeed, a gradient descent in the latent space needs to be performed. Hence, we will not consider this approach further in this work.

Adaptivity Neural network reconstructions can suffer from severe instabilities. This issue was notably discussed in [5], where it was shown that well chosen additive noise (an adversarial attack) or modifications of the forward operator could lead to disastrous hallucinations for some specific architectures. This problem was studied with care in [29]. There, the authors have shown that careful training procedures could fix this issue and yield robust and state-of-the-art reconstruction results.

A paper closely related to our work is [30]. The authors study the same robustness issue to model mismatches. The authors propose two distinct algorithmic approaches to attack it. The first one is called *parametrize & perturb* by the authors. It suffers from an important drawback, which is the need to re-optimize the network weights for every new operator. It can therefore be slow at runtime and we do not compare it in this paper. The other approach is called *Reuse & Regularize (R&R)*. It consists in training a network for a given operator $\mathbf{A}(\boldsymbol{\theta}_0)$, and then use this network for another operator $\mathbf{A}(\boldsymbol{\theta}_1)$. This is done in an iterative procedure, accounting for the data consistency term $\|\mathbf{A}(\boldsymbol{\theta}_1)\mathbf{x} - \mathbf{y}\|_2^2$. The approach we propose in this paper is significantly lighter at run-time, since we just train the network with a family of operators.

An older and popular alternative consists in replacing the proximal operator in (6) by a denoiser. This approach is often called a *plug&play (P&P)* prior [71]. It was first used with hand-crafted priors [36] and a significant performance boost occurred with the use of pre-trained neural networks among which we can cite [62, 80]. This approach has the huge asset of adapting painlessly to arbitrary inverse problems. We propose some comparisons and discuss the pros and cons of each approach in Section 6.

Blind inverse problems Blind inverse problems are spread massively in applications and it is impossible to provide a comprehensive overview of the existing works. The review papers [42, 14] provide a good idea of the wealth of results for the sole field of blind deconvolution and super-resolution.

A possibility is to design a two-step method. First an estimate of the forward operator is built. Second, this estimate is used in conjunction with the methods from the previous section. In some cases, it is possible to exploit some redundancy in the data to estimate the operator parameters. This is the

case in parallel Magnetic Resonance Imaging (MRI), where the coil sensitivity maps can be estimated using only the low frequencies [66, 57, 35]. When no redundancy is available, estimating the operator can be achieved by minimizing the discrepancy between the statistics of the acquired measurement and the statistics of the measurements generated by applying an operator to a “natural” signal. A good example in blind deblurring is the Goldstein-Fattal approach [31], which analyzes the power spectrum of the blurry image. Recently, a few authors proposed to build an identification network that learns to identify the blur kernel [63] or a blur parametrization [67, 77, 16, 22] from the blurry-noisy image. While this approach is cheap computationally, it requires an application specific design and we will not consider it further in this work.

One of the most popular alternatives consists in minimizing a combination of a data fidelity term and a regularizing prior. This can be addressed through an alternate minimization between the image and the operator parametrization. Most of the literature suggests the use of hand-crafted priors on the unknown operator or on the image to recover (see e.g. [17, 27, 40, 41, 76, 4, 53, 51, 54, 58, 9, 49, 19, 81] for blind deblurring, or [59, 74] in CT imaging).

While these approaches can provide excellent results, they are likely to be outperformed by neural network based approaches in a near future. Indeed, impressive performance has already been reached recently thanks to neural network based regularizers. Different strategies have been suggested, going from untrained networks (see [11] for an application in optics), generative models (see [8] for an application in blind deblurring), or unrolled networks (see [44, 45] for an application to super-resolution from an image sequence).

The method advocated in our paper is close in spirit to the works in this latest category. It differs in the way the training is performed. Here, we first train an unrolled network on a family of forward operators, which allows fixing the weights once for all. We then minimize (5) in the space of parameters of the forward model. This methodology has various advantages:

- Compared to untrained networks [11], the method does not optimize the network weights to solve the problem, which is typically quite heavy computationally. It is therefore faster at runtime. In addition, it is adapted to a clearly defined image dataset.
- Methods based on generative models [10, 8] may suffer from a significant drawback: the produced images necessary live in the range of the generator. To avoid this issue, a possibility is to add hand-crafted regularization terms such as total variation that allow extending the span of possible images [8].
- In [44, 45], the neural network weights are trained directly to solve the blind inverse problem. This significantly limit the number of weights and iterations within the iterative procedure. In this paper, we propose to train the network beforehand, allowing to use arbitrary solvers and as many iterations as desired to find the parameter θ .

3 Preliminaries

In this paper, we consider forward models

$$\mathbf{y} = \mathbf{A}(\theta)\mathbf{x} + \mathbf{b} \quad (7)$$

where $\mathbf{A}(\theta) \in \mathbb{K}^{M \times N}$ is a linear mapping either real ($\mathbb{K} = \mathbb{R}$) or complex ($\mathbb{K} = \mathbb{C}$). In our experiments, we consider additive white Gaussian noise (complex for MRI) $\mathbf{b} \sim \mathcal{N}(0, \sigma^2 \text{Id})$. The dependency of \mathbf{A} with respect to its parameter θ can be linear or nonlinear. We let $N \in \mathbb{N}$ denote the number of pixels of the image \mathbf{x} with $N = N_x \times N_y$ for 2D images and M is the number of measurements.

3.1 The forward models

To illustrate our problem, we consider three important biomedical applications: parallel magnetic resonance imaging, computerized tomography and microscopy/astronomy. Let us describe these applications more precisely.

Parallel Magnetic Resonance Imaging Our aim here is to reconstruct images from under-sampled Fourier samples with unknown sensitivity maps associated to $J \in \mathbb{N}$ reception coils, and with inaccurate trajectories. The parameter θ can be decomposed as $\theta = (\tau, \omega)$, where τ is the parameter describing the

sensitivity maps and ω describes a perturbation of the sampling locations. To the best of our knowledge, these two problems have not been treated jointly in the literature yet.

Let $\mathcal{F}(\xi)$ denote the non-uniform Fourier transform (NUFT) [55] at frequencies ξ , defined by

$$[\mathcal{F}(\xi)]_{m,n} = e^{-i\langle \mathbf{p}_n, \xi_m \rangle}$$

where $(\mathbf{p}_n)_{1 \leq n \leq N}$ is a set of 2D positions on a grid. We construct a family of forward operators $\mathcal{A} = \{\mathbf{A}(\xi, \theta), \xi \in \Xi, \theta \in \Theta\}$, where $\Xi \subset \mathbb{R}^{2 \times M}$ is a set of 2D sampling schemes with M sampling points. The parameter space $\Theta = \mathcal{T} \times \Omega$ describes the set of admissible parameters for the sensitivity maps \mathcal{T} and for the perturbation Ω . The measured signal $\mathbf{y} = (\mathbf{y}^{(1)}, \dots, \mathbf{y}^{(J)})$ is acquired through J coils. The m -th measurement acquired by the j -th coil is defined by

$$y_m^{(j)} = [\mathbf{A}(\xi, \theta)\mathbf{x}]_{m,j} + \mathbf{b}_{m,j} = \left[\mathcal{F}(\mathbf{h}(\omega) \star \xi) \left(\mathbf{x} \odot \mathbf{s}(\tau^{(j)}) \right) \right]_m + \mathbf{b}_{m,j}. \quad (8)$$

The mapping $\mathbf{s} : \tau^{(j)} \in \mathbb{R}^T \mapsto \mathbf{s}(\tau^{(j)}) \in \mathbb{C}^N$ parametrizes the coil sensitivity maps. Since the sensitivity maps are smooth, we use a parametrization based on thin plate splines (TPS) [26]. The total number of parameters that encode the sensitivity map is $T = 104$. It consists of the TPS coefficients using 7×7 regularly spaced control points plus the coefficients of a first degree polynomial. This has to be multiplied by two for the real and imaginary parts.

Following [70, 24], we assume that the trajectory ξ is perturbed by a convolution with an impulse response $\mathbf{h}(\omega)$. The symbol \star in (8) corresponds to a discrete convolution. Evaluating the convolution filter $\mathbf{h}(\omega)$ is known as a challenging problem that can be addressed with (expensive) field cameras [24]. Here, in the spirit of [70], we will rather treat it as a blind inverse problem. We parametrize \mathbf{h} as a linear combination of the form $\mathbf{h}(\omega) = \sum_{o=1}^O \omega_o \mathbf{h}_o$, where $(\mathbf{h}_o)_{1 \leq o \leq O}$ is an orthogonal basis. In practice, we simply use compactly supported filters of size $O = 32$ and $(\mathbf{h}_o)_{1 \leq o \leq O}$ corresponds to the first 32 elements of the canonical basis.

Computerized Tomography Our aim is to reconstruct images from parallel beam computerized tomography. The parameter θ describes the projection angles (or the patient motion).

We assume that the CT scan uses parallel beams and that it performs J acquisitions with a receptor that has M sensors, resulting in $J \times M$ measurements. In this application, the parameter $\theta = (\alpha, \mathbf{s})$ represents the angles $\alpha \in \mathbb{R}^J$ and the shifts at the origin $\mathbf{s} \in \mathbb{R}^J$ that describe the beams trajectories. If m corresponds to the m -th pixel of the receptor and if we index the acquisitions by $1 \leq j \leq J$, we get

$$y_m^{(j)} = \iint_{\Omega} \mathbf{x}(u_x, u_y) \delta_{u_x \cos(\alpha_j) + u_y \sin(\alpha_j) = \mathbf{p}_m + \mathbf{s}_j} du_x du_y + b_m^{(j)}, \quad (9)$$

with $\Omega = [-N_x/2, N_x/2] \times [-N_y/2, N_y/2]$ and $\mathbf{p} = \llbracket -M/2, \dots, M/2 - 1 \rrbracket$. A perfect model would correspond to α being equispaced angles and $\mathbf{s} = \mathbf{0}$. The forward model can be computed using the Fourier slice theorem. This corresponds to performing a 2D NUFT and we resort to the same library used as for MRI (see <https://github.com/albangossard/Bindings-NUFFT-pytorch>).

Deblurring in optics In this application, we wish to solve problems appearing in optics, especially microscopy or astronomy. The parameter θ describes the point spread function through the theory of diffraction. The acquisition model in this application simply reads

$$\mathbf{y} = \mathbf{h} \star \mathbf{x} + \mathbf{b}, \quad (10)$$

where \mathbf{h} is the kernel blur. We consider blurs generated by Fresnel diffraction theory [32]. The kernel blur is parametrized by a vector $\theta \in \mathbb{R}^7$ and the convolution kernel is expressed as

$$\mathbf{h}(\theta) = c \left| \int_{\|w\|_2 \leq f_c} \exp \left(2i\pi \left[\sum_{k=1}^K \theta_k Z_k + \langle \mathbf{u}, \mathbf{w} \rangle \right] \right) dw_b \right|^2. \quad (11)$$

In this expression, f_c is a cutoff frequency and c is a scaling parameter such that $\|\mathbf{h}\|_1 = 1$. The expansion $\sum_{k=1}^K \theta_k Z_k$ describes the pupil function of an objective. The functions Z_k are Zernike polynomials and the vector θ therefore parametrizes the pupil function.

3.2 The model-based reconstruction networks

We consider three convolutional neural networks all based on a fixed convolutional neural network architecture \mathcal{D} which is a DruNet network [80]. This network is the current state-of-the-art when used within plug&play algorithms. One of its important assets is its ability to accommodate for different noise levels. The idea is to set one of the input channels as a constant image with a value equal to the standard deviation of the noise.

3.2.1 Denoising network

The denoising network is denoted \mathcal{N}^d . It is of the form

$$\mathcal{N}^d[\mathbf{w}, \boldsymbol{\theta}, \mathbf{y}] = \mathcal{D}[\mathbf{w}, \mathbf{A}(\boldsymbol{\theta})^\dagger(\mathbf{y})].$$

3.2.2 Unrolled proximal gradient network

Letting $F(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}(\boldsymbol{\theta})\mathbf{x} - \mathbf{y}\|_2^2$, the unrolled proximal gradient descent takes the sequential form:

$$\begin{aligned} \mathbf{x}_0 &= \mathbf{A}(\boldsymbol{\theta})^\dagger \mathbf{y} \\ \mathbf{x}_{k+1} &= \mathcal{D}[\mathbf{w}_k, \mathbf{x}_k - \gamma \nabla F(\mathbf{x}_k)]. \end{aligned}$$

where $\gamma = \frac{1}{\|\mathbf{A}(\boldsymbol{\theta})\|_{2 \rightarrow 2}^2}$ is a step-size. The reconstruction network runs for K iterations and is denoted $\mathcal{N}^p : (\mathbf{w}, \boldsymbol{\theta}, \mathbf{y}) \mapsto \mathbf{x}_K$.

3.2.3 Unrolled ADMM

It takes the form (see e.g. [68]):

$$\begin{aligned} \mathbf{x}_0 &= \mathbf{A}(\boldsymbol{\theta})^\dagger \mathbf{y} \quad \text{and} \quad \boldsymbol{\mu}_0 = 0 \\ \mathbf{z}_{k+1} &= [\mathbf{A}(\boldsymbol{\theta})^* \mathbf{A}(\boldsymbol{\theta}) + \beta \text{Id}]^{-1} (\mathbf{A}(\boldsymbol{\theta})^* \mathbf{y} + \beta \mathbf{x}_k - \boldsymbol{\mu}_k) \\ \mathbf{x}_{k+1} &= \mathcal{D} \left[\mathbf{w}_k, \mathbf{z}_{k+1} + \frac{\boldsymbol{\mu}_k}{\beta} \right] \\ \boldsymbol{\mu}_{k+1} &= \boldsymbol{\mu}_k + \beta (\mathbf{z}_{k+1} - \mathbf{x}_{k+1}). \end{aligned}$$

This sequence runs for K iterations and the result is denoted $\mathcal{N}^a : (\mathbf{w}, \boldsymbol{\theta}, \mathbf{y}) \mapsto \mathbf{x}_K$. The parameter β is a penalty parameter, which is fixed in our experiments.

In the two unrolled algorithms, the weights \mathbf{w} to be trained are $\mathbf{w} = [\mathbf{w}_0, \dots, \mathbf{w}_{K-1}]$ and they are not shared across iterations.

4 Training on a family of operators

Traditionally, networks are trained by minimizing the empirical risk for a given forward model $\boldsymbol{\theta}_0$. Our first contribution is to train *end-to-end* networks by minimizing the risk over a set of forward operators:

$$\inf_{\mathbf{w} \in \mathbb{R}^D} \frac{1}{2} \mathbb{E} \|\mathcal{N}[\mathbf{w}, \boldsymbol{\theta}, \mathbf{y}] - \mathbf{x}\|_2^2, \quad (12)$$

where the expectation is taken with respect to the noise \mathbf{b} , to the images \mathbf{x} and to the forward models $\boldsymbol{\theta}$.

4.1 What is different?

Let us provide a rough theoretical explanation of the difference between a training on a single operator and an operators family. To this end, we focus on the simplest denoising network. The pseudo-inverse $\mathbf{A}(\boldsymbol{\theta})^\dagger$ applied to $\mathbf{y} = \mathbf{A}(\boldsymbol{\theta})\mathbf{x} + \mathbf{b}$ yields a vector $\hat{\mathbf{x}}$ of the form

$$\hat{\mathbf{x}} = \mathbf{x} + \mathbf{k} + \mathbf{n}, \quad (13)$$

where $\mathbf{k} \in \ker(\mathbf{A}(\boldsymbol{\theta}))$ and \mathbf{n} is a correlated Gaussian noise living in $\text{ran}(\mathbf{A}(\boldsymbol{\theta}))$. Hence the denoising network \mathcal{D} serves two purposes: 1) recover the missing data \mathbf{k} in the kernel of $\mathbf{A}(\boldsymbol{\theta})$ and 2) remove

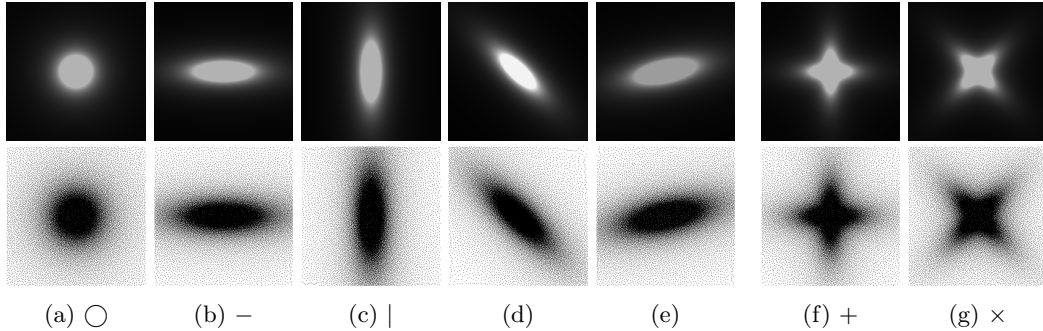


Figure 2: Densities (top) and corresponding sampling schemes (bottom). Fig. 2a, 2b, 2c, 2d and 2e belong to the family \mathcal{X} . Fig. 2f and 2g (crosses) do not. Notice that the sampling patterns are diverse with significant differences from one to the other.

the correlated noise \mathbf{n} . Each of these two tasks is clearly highly dependent on θ . If trained with a single scheme, the network may get specialized very well for these specific statistical patterns and not extrapolate to other ones. We will explore in the numerical section 6.2, whether a training with a larger variety of operators mitigates the performance drop.

4.2 Choosing distributions of operators

In this section, we focus on designing distributions for θ , for the different applications listed above.

4.2.1 Magnetic Resonance Imaging

In this modality, the family of forward operators is constructed by considering different sampling schemes, sensitivity maps and trajectory perturbations.

Sampling schemes We propose to generate random sampling schemes ξ following the ideas from [12, 18, 43]. The principle is to design a scheme that fits a target probability measure $\rho : \mathbb{R}^2 \rightarrow \mathbb{R}_+$. To this end, we define

$$\xi(\rho) \stackrel{\text{def}}{=} \arg \min_{\xi \in \Xi} \text{dist} \left(\frac{1}{M} \sum_{m=1}^M \delta_{\xi_m}, \rho \right), \quad (14)$$

where dist is a discrepancy between probability measures and $\Xi \subseteq \mathbb{R}^{2 \times M}$ is a set that describes the admissible trajectories from the scanner.

Following [33], we generate random target densities ρ as anisotropic power decaying distributions. They are parametrized by a random vector λ that encodes the density at origin, the anisotropy and the power decay law. To avoid solving (14) at training time, we have pre-computed 1000 sampling patterns. The corresponding vectors λ have been generated by using a max-min sampling (see [56, 22]) of a set of an admissible set of parameters Λ . We refer to [33] for more details. Examples of densities and sampling patterns $\xi(\rho(\lambda))$ are displayed in Fig. 2a-2e.

Sensitivity maps As for the sensitivity maps, we used real estimates generated using the fastMRI database [78]. We first estimate them using a standard approach [35] and then project the estimates onto the span of the proposed parametrization with thin plate splines. At training time, they are associated to the corresponding training pairs.

Trajectories filtering We did not include the trajectory perturbation effect (convolution with $\mathbf{h}(\omega)$) at training time.

4.2.2 Computerized tomography

In this modality, we assume that the distribution of projection angles follows a uniform distribution centered on a vector of regularly spaced angles $\alpha_0 = (-\pi/4, -\pi/4 + \pi/J, \dots, \pi/4)$ (see the red lines in first row of Fig. 1) and shift at origin $\mathbf{s}_0 = 0$. We only sample the angles in the range $[-\pi/4, \pi/4]$ to account for the missing cone in computerized tomography.

Hence, we have $\boldsymbol{\alpha} = \boldsymbol{\alpha}_0 + \boldsymbol{\alpha}_\delta$ with $\boldsymbol{\alpha}_\delta \sim \mathcal{U}([-1.37^\circ, 1.37^\circ]^J)$ and the random shifts are $\mathbf{s} \sim \mathcal{U}([-2, 2]^J)$. These perturbations may reflect movements of the patient inside the scanner during the scan.

4.2.3 Deblurring

In this application, we vary the blur kernel by changing only the 4-th to the 10-th Zernike polynomials. We set $\theta_1 = \theta_2 = \theta_3 = 0$ in (11) and we let the coefficients θ_4 to θ_{10} follow a uniform distribution in $[-0.15, 0.15]$.

5 Solving blind inverse problems

After the training procedure on a family of operators, we get a reconstruction mapping $\mathcal{N}[\mathbf{w}^*, \boldsymbol{\theta}, \mathbf{y}]$. Now, if $\mathbf{y} = \mathcal{P}(\mathbf{A}(\bar{\boldsymbol{\theta}})\mathbf{x})$, with an unknown parameter $\bar{\boldsymbol{\theta}}$, we propose to solve the optimization problem (5). As the function in (5) is deterministic over a small to moderate dimension, we can opt for basically any standard optimization routine. The use of automatic differentiation techniques available in PyTorch or Tensorflow allows us to compute the Jacobian of $\mathcal{N}[\mathbf{w}^*, \boldsymbol{\theta}, \mathbf{y}]$ with respect to the parameter $\boldsymbol{\theta}$. Possible optimization routines include:

- The L-BFGS optimizer [48]. This quasi-Newton method estimates the Hessian of the function using first order information only and is known to converge rapidly when initialized close to a (local) minimizer. It therefore seems particularly adapted when the user has a good knowledge of the true parameter $\bar{\boldsymbol{\theta}}$.
- The RMSProp or ADAM optimizers [69, 39]. For the computerized tomography and blind deblurring problems, we observed significant issues with a convergence to bad local minimizers only. To avoid this phenomenon, a possibility is to resort to inertial methods, which are known to escape narrow basins of attraction. In our experiments, we used the RMSProp optimizer with a parameter $\beta = 0.9$ (Adam with $\beta_1 = 0$).
- Global optimization. If it turns out that the cost function is too chaotic, the gradient of the objective function does not provide a meaningful information on the location of the global minimizer. We can then resort to 0-th order methods such as Bayesian optimization [28].

As is, the prior on the forward model is encoded by the physics of the acquisition system through the mapping $\mathbf{A}(\cdot)$. Hence, we add no regularization term on $\boldsymbol{\theta}$. It would also be possible to add one to promote specific solutions. This is the standard approach in Bayesian estimation. We did not explore this idea in this paper.

6 Numerical experiments

The numerical experiments are divided in two sections. In the first section 6.2 we compare the benefits and drawbacks of training model-based networks on a family of operators. We consider the training of a denoising network and of an unrolled proximal gradient network for MR image reconstruction. The experiments are conducted in a simplified framework where the constraints and the sensitivity maps are not taken into account.

In the second section 6.3, we illustrate that training model-based networks on a family of operators allows solving blind inverse problems. The experiments are carefully conducted on the three applications: MRI, CT and image deblurring with an unrolled ADMM [68] with $K = 5$ iterations. The MRI experiments in this part take into account both the constraints and the sensitivity maps.

6.1 The setting

All the models were trained using the Adam optimizer in PyTorch with the default parameters except the learning rate which was tuned for each experiment.

Magnetic Resonance Imaging The training database is the fastMRI knee training dataset [78]. It contains 34,742 images of size 320×320 . All evaluations were performed on the validation set of the fastMRI knee database containing 7,135 2D slices. We used the efficient cuFINUFFT transform [64], which is the fastest available library in our experiments (see <https://github.com/albangossard/Bindings-NUFFT-pytorch> for comparisons).

For the experiments illustrating the advantages of training on a family of operators, we set $M = N/4$, i.e. a 4x downsampling rate. We used a single reception coil ($J = 1$) with a known sensitivity map $\mathbf{s} = 1$. The denoising network \mathcal{N}^d was trained on 30 epochs with a learning rate of 10^{-3} and an exponential step decay of 0.95 after each epoch. The unrolled network \mathcal{N}^p uses $K = 10$ iterations and it was trained on 14 epochs with a learning rate of 10^{-4} and an exponential step decay of 0.95 after each epoch. Both trainings took about 24h on an Nvidia V100, resulting in a total energy of ~ 70 kWh.

The blind reconstruction experiments are conducted with $M = N/10$ measurements and $J = 15$ reception coils. The networks are trained for 8 epochs with a learning rate of 10^{-4} and with an exponential step decay of 0.95 after each epoch.

Computerized Tomography The dataset used is the Lung Image Database Consortium [6] which has a total of 244,527 slices. We divided this dataset into a training and a validation dataset (80% and 20% respectively). As the blind inverse problem (5) requires differentiating the operator $\mathbf{A}(\boldsymbol{\theta})$ with respect to its parameters $\boldsymbol{\theta}$, we cannot use standard GPU-based libraries to compute the Radon transform [60]. We thus resorted to an homemade implementation that relies on a NUFT through the Fourier slice theorem. In order to reduce the important numerical cost and energy consumption of the experiments with CT reconstruction, we downsized the images to 256×256 .

Deblurring The image deblurring experiments use the MS COCO dataset [47] (118,287/5,000 images for training/validation). During training we randomly cropped patches of size 400×400 to accelerate the computation.

6.2 The benefits of training on a family in MRI

In this section, we show the advantages and drawbacks of training a reconstruction network on a family of operators. In this version of the work, we only display the results related to MRI with a single coil. We plan to conduct similar experiments for all imaging modalities for the final version of this work.

6.2.1 Training on fixed sampling schemes

In this section, we trained the two reconstruction networks (the denoising network \mathcal{N}^d and the unrolled proximal gradient descent \mathcal{N}^p) on 5 different schemes: a radial one (\circ , Fig. 2a), an horizontal one ($-$, Fig. 2b) and a vertical one ($|$, Fig. 2c). In addition, we used two crosses, which do not belong to the training family \mathcal{X} . The first one is aligned with the axes ($+$, Fig. 2f) and the other one with the diagonals (\times , Fig. 2g). In Table 1, we report the average peak signal-to-noise ratio, on the validation set for the two architectures. Table 1 illustrates various facts listed below.

Lack of adaptivity Without surprise, the values on the diagonal are higher than the off-diagonal terms. This means that the best way to reconstruct images for a given scheme is to train the network for this specific scheme. The drop of peak signal-to-noise ratio (PSNR) when using a network trained with the wrong operator can be as high as 9dB for the denoising net and 5dB for the unrolled net (see the pairs $-$ and $|$). This is a striking illustration of the strong dependency of a reconstruction net to the operator used at the training stage. The corresponding images are shown in Fig. 3.

The superiority of unrolled nets The unrolled network provides better reconstruction results than the denoising net. The overall gain on the diagonal varies between 1.4dB and 1.7dB for this particular application, which is significant. This is in accordance with recent comparisons of both strategies [52].

Partial adaptivity While the training on the vertical scheme $|$ provides catastrophic results when used on the horizontal scheme $-$, the networks trained on the radial \circ scheme provide rather good results uniformly on the 4 other sampling schemes. Indeed, we see that the performance drop when used on a distinct scheme raises up to 3.6dB for the denoising net and 2.7dB for the unrolled network. Overall, the unrolled network provides a significantly better alternative when it comes to adaptivity.

		Train					Train				
		○	-		+	×	○	-		+	×
Evaluation	○	36.30 ±4.30	31.54 ±2.62	33.15 ±2.91	35.52 ±3.71	35.19 ±3.58	38.04 ±5.13	37.17 ±4.60	36.72 ±4.48	37.91 ±5.05	37.87 ±5.02
	-	32.93 ±2.79	35.43 ±3.96	28.27 ±2.54	33.45 ±3.01	31.32 ±2.62	35.93 ±4.10	37.09 ±4.64	32.97 ±2.60	35.65 ±4.03	35.38 ±3.64
		32.68 ±2.84	26.71 ±2.50	36.20 ±4.20	34.31 ±3.20	30.15 ±2.62	35.37 ±4.09	32.13 ±3.06	37.61 ±4.82	37.09 ±4.53	36.50 ±4.18
	+	35.36 ±3.65	29.76 ±2.57	32.10 ±2.94	36.34 ±4.28	32.83 ±2.74	37.77 ±4.92	36.09 ±4.09	36.41 ±4.02	37.98 ±5.03	37.32 ±4.60
	×	35.21 ±3.63	32.77 ±2.72	33.01 ±2.75	33.77 ±2.95	36.04 ±4.19	37.56 ±4.90	36.96 ±4.47	35.74 ±4.17	36.96 ±4.50	37.74 ±4.98

(a) Denoising net

(b) Unrolled net

Table 1: Average PSNR and its standard deviation evaluated on the fastMRI validation dataset for the two architectures.

		Model					
		ID	FD	IU	FU	P&P	R&R
Evaluation	○	36.30 ±4.30	36.03 ±4.10	38.04 ±5.13	38.00 ±5.09	35.06 ±3.53	35.20 ±3.56
	-	35.43 ±3.96	35.06 ±3.74	37.09 ±4.64	36.97 ±4.57	33.95 ±3.25	32.88 ±2.92
		36.20 ±4.20	35.89 ±3.99	37.61 ±4.82	37.52 ±4.77	35.12 ±3.54	34.21 ±3.23
	+	36.34 ±4.28	35.35 ±3.71	37.98 ±5.03	37.88 ±4.96	35.18 ±3.56	34.13 ±3.17
	×	36.04 ±4.19	35.18 ±3.59	37.74 ±4.98	37.66 ±4.92	34.66 ±3.41	33.57 ±3.04

Table 2: Average PSNR and standard deviation (in dB) of various reconstruction approaches applied to various operators.

Optimal sampling scheme Some sampling schemes make the reconstruction easier than others, which is in accordance with the compressed sensing theory. The knee images have large vertical and horizontal edges. This seems to favor the + sampling scheme, which was already observed in other works [73, 75, 33].

6.2.2 Training on an operator family

In this section, we trained the reconstruction networks by varying the forward operators, as in (12). In what follows, we let ID and IU denote the “ideal” denoising network and the “ideal” unrolled network respectively. By ideal, we mean that the networks have been trained and tested with the same operator. They serve as a benchmark that cannot be outperformed. We let FD and FU denote the “family” denoising and unrolled networks, which have been trained over a complete family. We also tested the plug&play approach. We used an unrolled proximal gradient for $K = 10$ iterations with a DruNet network as an embedded denoiser. It was trained specifically to denoise the images with various levels of white Gaussian noise. Finally, we implemented the regularize&reuse network (R&R) [30] composed of $K = 10$ iterations. The embedded inversion network consists of a pseudo-inverse, followed by a DruNet network trained for the ○ sampling scheme. The hyperparameters in the method (see [30]) were tuned to produce the best results. Table 2 shows the performance of the different architectures. The following conclusions can be drawn.

Denoising network The denoising network trained on the whole family drops by less than 0.4dB compared to ID, when tested with schemes that belong to the training family.

However, the performance is altered (1dB) for the two schemes outside the training family. This suggests that a proper training of a denoising net should cover a sufficiently vast family of operators.

Unrolled networks The unrolled networks trained with a family shows a performance drop of at most 0.12dB compared to IU for schemes inside *and outside* the family. This is a remarkable feature: the scheme is able to extrapolates outside the training set to some extent. This illustrates one of the take home message of our paper: training unrolled networks on a family does not degrade much the performance while providing adaptivity of the networks.

In addition, the price of adaptivity seems affordable for most applications since a 0.12dB loss is marginal.

Plug & Play (P&P) When comparing Table 1 and 2, we see that the plug&play approach performs better uniformly than models trained on a single operator. However, its performance drops by 1 – 1.5dB compared to ID and 2.5 – 3.1dB compared to IU.

It is also significantly less accurate than FU for an identical computational cost. This suggests that for a given reconstruction architecture, it is beneficial to train the proximal networks for a specific task rather than using a *universal* denoiser, as is the case in plug&play.

Notice however, that FU does not extrapolate well to problems completely different from the ones it was trained for. Indeed, we trained FU for an MRI reconstruction problem and tested it for a deblurring application. There, the plug&play approach was considerably more consistent. In a sense, we can see the proposed training as an intermediate step between the plug&play approach (adaptable to all inverse problems) and the traditional training of reconstruction networks (perfectly adapted to a single operator).

Reuse & Regularize (R&R) Finally, the R&R approach does improve the results by up to 1.5dB compared to a model trained on a single operator. However, it seems that our simpler training approach provides significantly better results. Notice that R&R has a wider scope since it also deals with unknown perturbations of the forward operator (i.e. blind inverse problems), while we only consider the non blind case in this section.

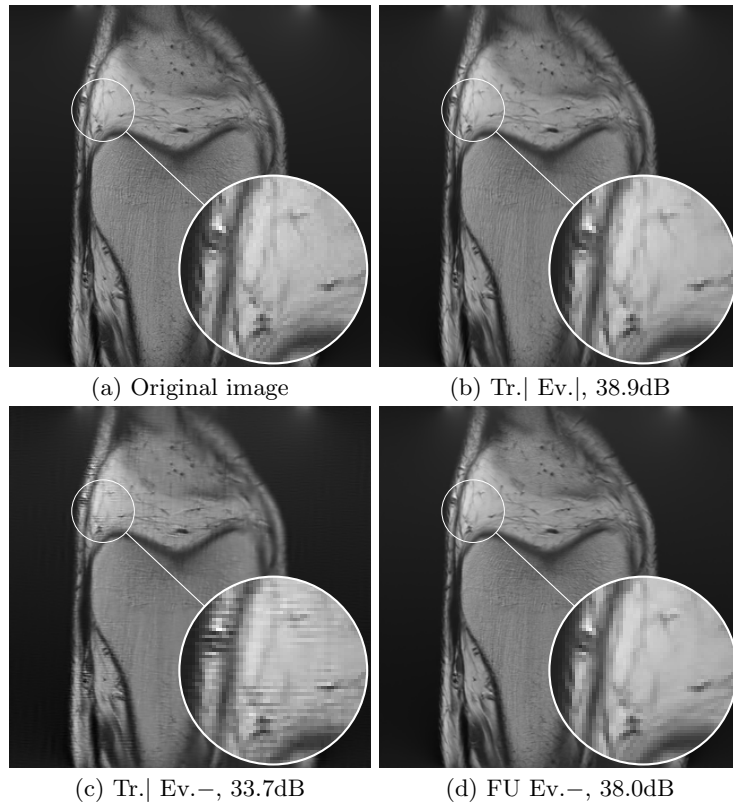


Figure 3: Examples of reconstructions using the unrolled network. We trained it on $|$ (3b, 3c) and on a family (3d). We tested it on $|$ (3b) and $-$ (3c, 3d).

6.3 Blind inverse problems

In this section, we illustrate how training on a family of operators allows us to solve different blind inverse problems by minimizing (5). Fig. 4, 6 and 5 illustrate some of the results for MRI, deblurring and CT imaging respectively. In these experiments, we assume that

$$\mathbf{y} = \mathbf{A}(\boldsymbol{\theta}_1)\mathbf{x} + \mathbf{b}, \quad (15)$$

for some unknown parameter $\boldsymbol{\theta}_1$ describing the forward model. Starting from an initial guess $\boldsymbol{\theta}_0$, we then solve (5) with a first order method, resulting in an estimate $\hat{\boldsymbol{\theta}}_1$ of $\boldsymbol{\theta}_1$. Fig. 4, 5, 6 show the performance of the solver for various applications. Let us analyze these results.

6.3.1 Magnetic Resonance Imaging

This application provides the most impressive results for various reasons:

- To the best of our knowledge, no one yet attempted to estimate the sensitivity maps and trajectory errors jointly. Estimating divergence in trajectories might look hopeless at first sight, which may explain this fact. Indeed, looking at the differences between $\boldsymbol{\xi}_1$ and $\boldsymbol{\xi}_0$ (see top-right and the zoom on the right-most column of Fig. 4) we see that the frequency shifts are huge (up to 5 pixels).
- The total number of parameters to estimate is really large. Indeed, it consists in the 104×15 parameters describing the sensitivity maps and the 32 parameters describing the convolution kernel that perturbs the trajectories, i.e. 1592 parameters.

If solved without any correction, the reconstruction results are disastrous (see the 2nd column). Solving the consistency problem (5) provides near perfect estimates of $\hat{\boldsymbol{\theta}}$ for all reconstruction mappings. For instance, the green $\hat{\boldsymbol{\xi}}_1$ and orange $\boldsymbol{\xi}_1$ trajectories cannot be distinguished on the right column. This may come as a surprise, and seems to suggest that this particular blind inverse problem is not as hard as may seem at first sight. This might be due to multiple redundancies in the data: the 15 reception coils associated to a slight oversampling of the k -space center (all the trajectories start exactly from the center) seem to ensure the identifiability of the problem. A nice research perspective is to explain this phenomenon from a theoretical viewpoint.

The reconstruction result obtained with the neural network trained on a family is significantly better than the two other ones (more than +1.3dB compared to the one trained on $\boldsymbol{\theta}_0$ and to the plug&play approach). In particular, the bone texture is reconstructed with the proposed approach, while it is not for the two others.

6.3.2 Computerized tomography

In this application, recall that the angle differences might be due to the motion of a patient in the scanner. As can be seen on the 2nd column, not accounting for this results in severe artifacts including some details loss and blur. This can be easily seen through the PSNR drop of more than 10dB from the 1st to the 2nd column.

A second observation is that the plug&play approach does not work for this application. The reason is likely the missing cone problem. The unrolled network is able to “learn” the invisible [13], that is to recover the kernel part \mathbf{k} in (13). In contrast, a neural network that would be trained only to remove additive Gaussian noise seems unable to do so. We see a clear advantage of the unrolled networks for this particular application.

The unrolled networks trained on the single operator $\boldsymbol{\theta}_0$ seems unable to recover the angles tilts: the average error only goes from 0.12° to 0.09° . On its side, the unrolled network trained on a family provides promising results, reducing the average tilts error to 0.024° . Recall that this training family consists of operators with random perturbations of the angles and positions. Both network correctly recover the shifts \mathbf{s}_1 , by reducing their size by a factor larger than 10. The gain in performance by training on a family is significant with a difference of more than 2.3dB for the proposed approach.

6.3.3 Blind deblurring

This application is one of the most studied in the literature, especially in computer vision. It has an important peculiarity: the basic block of the convolutional neural network is identical to the forward operator. Hence, when an unrolled network is trained, we can expect the networks $\mathcal{D}[\mathbf{w}_k, \cdot]$ to not only act as “denoisers”, but also as deconvolution mappings.

Known operator This fact might explain the second row of Fig. 6, where the unrolled network actually degrades the image quality compared to the observation. We indeed go from 21.05dB to 15.62dB with a known operator! This also confirms the conclusions of the introductory example in Fig. 1.

The behavior of the unrolled network trained on a family is significantly more appealing with a gain of 5.6dB compared to the observation when knowing the true operator. The plug&play provides a significant increase of 2.6dB, which is not on par with the performance of the proposed approach.

Unknown operator When the operator is unknown, only the proposed approach is able to recover an approximate version of the blur kernel (12dB compared to less than 4.5dB for the other approaches). The estimate is not perfect, but it is still sufficient to significantly improve the image resolution (21dB to 25.2dB), with clearly enhanced details (see e.g. the railway).

6.3.4 Additional experiments

Finally, to show that the approach is robust and versatile, we provide a few extra experiments in Fig. 7, 8 and 9.

7 Conclusion

In this work we studied the stability of model-based reconstruction networks to variations of the acquisition operator. We first illustrated significant performance drops when training the models on a single forward model. We then demonstrated on a realistic MRI reconstruction problem that a simple solution to mitigate this effect and ensure a good adaptivity is to train the model on a family of operators. It opens new interesting perspectives for computational imaging. A recent trend consists in optimizing the forward model and the reconstruction algorithm jointly (see e.g. [75, 34, 73] for examples in MRI). With a reconstruction method capable of adapting to a vast family of operators, it becomes possible to restrict the attention to the optimization of the forward model only [33].

We also showed that the design of networks being able to adapt to different forward models, allows solving a variety of blind inverse problems in a convincing way. This aspect is critical in most applications: the forward model is usually known at best approximately, and can even be completely unknown. The benefits of handling this issue are huge in terms of resolution and signal-to-noise-ratio gains.

This promising work opens many perspectives. First, our experiments were conducted with simulated measurements which can lead to both an inverse crime [20] and a data crime [65]. Extensive validation should be carried out on real imaging devices. Another question that is not addressed in this paper is how the proposed blind inverse solver behaves if the true operator is not in the range of the admissible operators $\theta \mapsto \mathbf{A}(\theta)$? This question is left for future work as well. Another interesting perspective would be to add motion correction for MRI. A motion in the image domain translates to a phase modulation in the Fourier domain. This is a critical issue in practice. The disconcerting ease with which we solved the estimation of trajectory shifts and sensitivity maps, sparks good hopes to solve this long resisting problem. Finally, a current weakness of the proposed approach, is a rather high computational complexity. Indeed, the model needs to be differentiated from 10 to 1000 times with respect to the parameter θ . In practice, this took from 10 minutes to 1 hour in the experiments carried out in this paper. This might be incompatible with real large scale applications. Hence, better optimization strategies should be developed.

Acknowledgement

This work was supported by the ANR Micro-Blind and by the ANR LabEx CIMI (grant ANR-11-LABX-0040) within the French State Programme “Investissements d’Avenir”. P. Weiss acknowledges the support of AI Interdisciplinary Institute ANITI funding, through the French “Investing for the Future—PIA3” program under the Grant Agreement ANR-19-PI3A-0004. This work was performed using HPC resources from GENCI-IDRIS (Grant 2021-AD011012210R1).

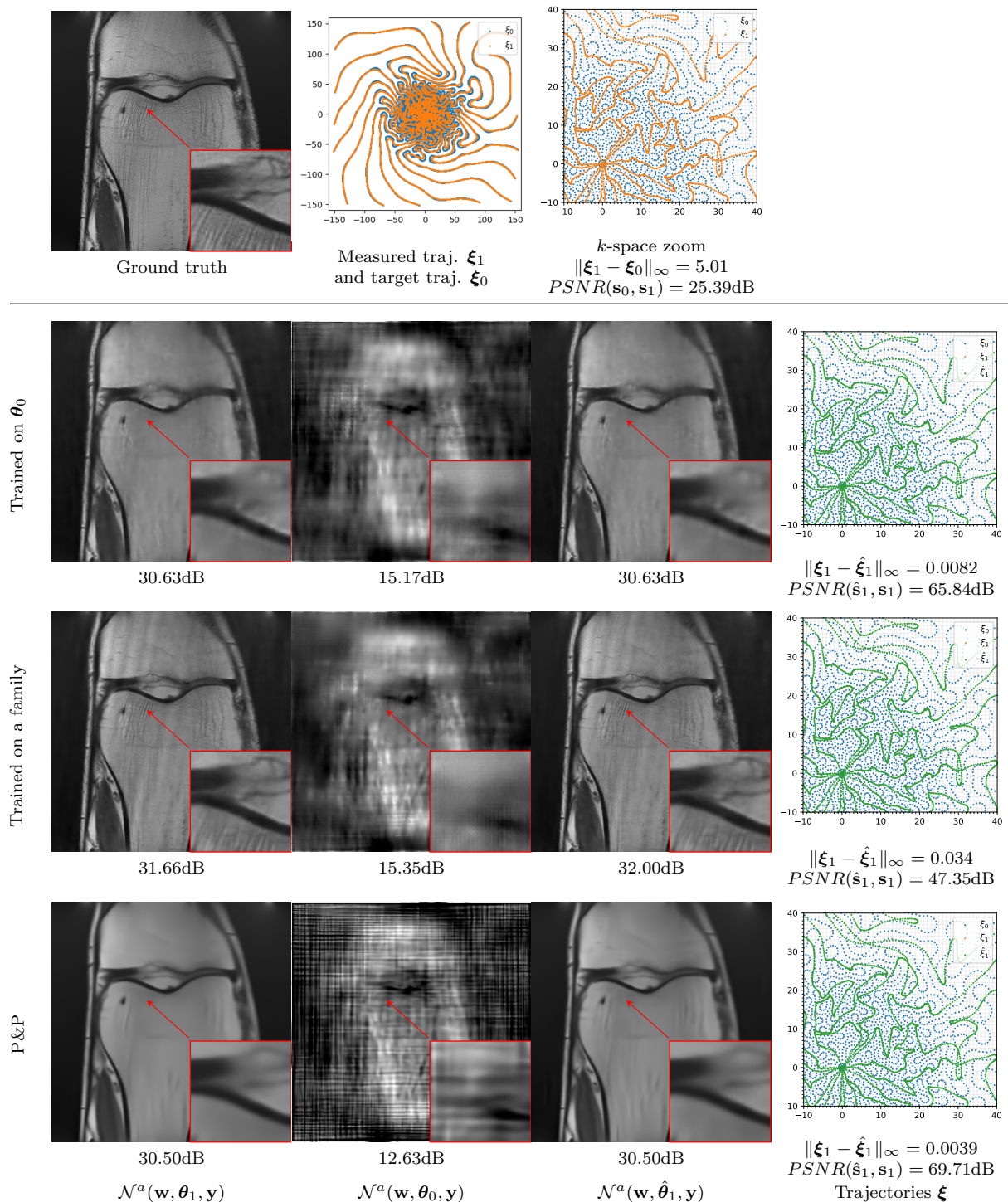


Figure 4: Self-calibrated MRI. *1st column*: reconstruction with a perfect knowledge of the forward model θ_1 . *2nd column*: reconstruction assuming the wrong forward model θ_0 . *3rd column*: reconstruction using the estimated forward model $\hat{\theta}_1$. *4th column*: estimate of the operator. We display the maximal distance between sampling points $\|\xi_1 - \hat{\xi}_1\|_\infty$ as well as the PSNR of the estimated sensitivity maps \hat{s}_1 . From top to bottom: different training strategies are compared. *2nd row*: trained on θ_0 . *3rd row*: trained on a family of operators. *4th row*: using a plug&play prior. The PSNR is indicated below each image. The noise level given to the P&P denoiser has been tuned as to yield the best PSNR on the non-blind problem. Other models do not require tuning at evaluation.

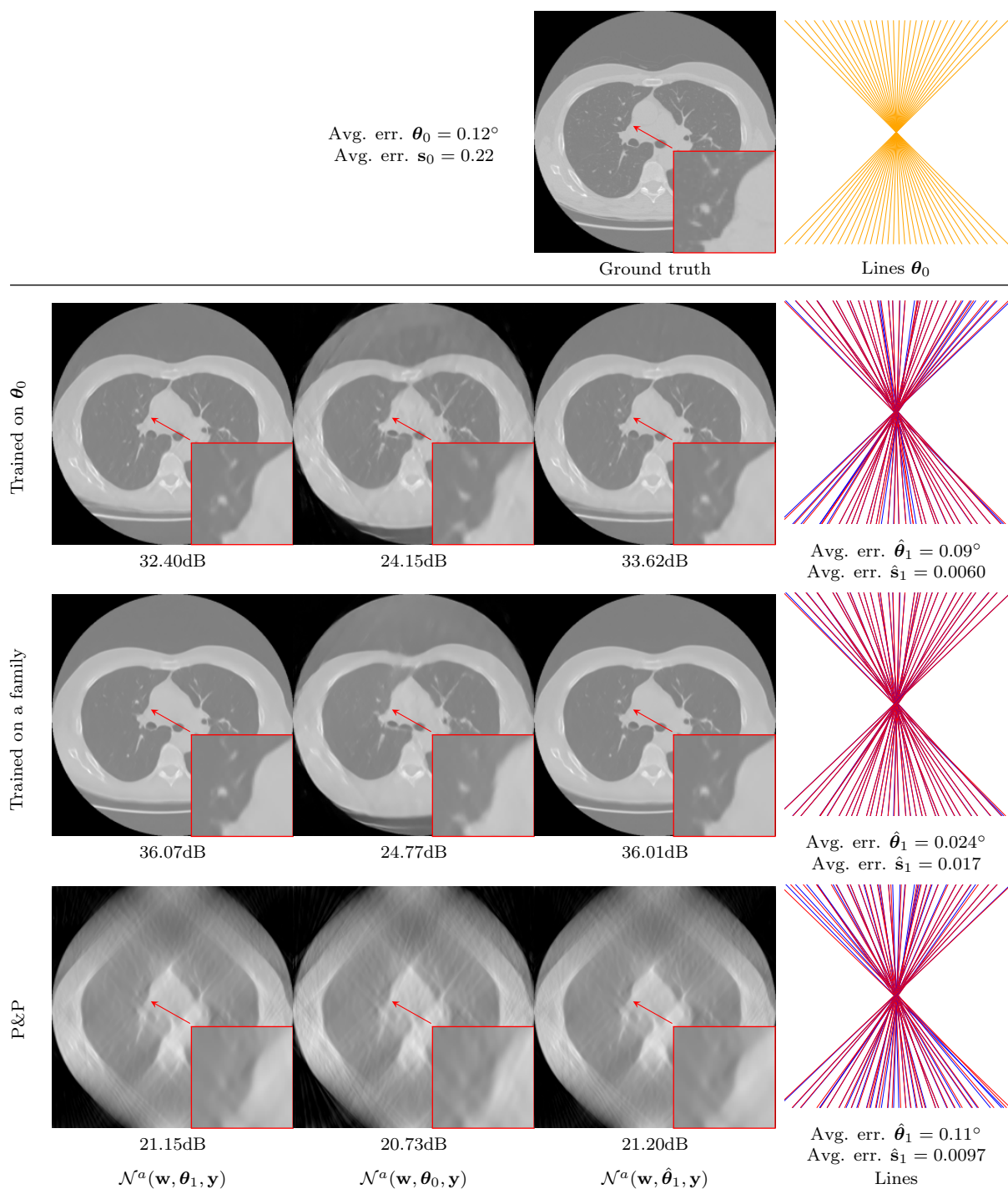


Figure 5: Self-calibrated computerized tomography. *1st column*: reconstruction with a perfect knowledge of the forward model θ_1 . *2nd column*: reconstruction assuming the wrong forward model θ_0 . *3rd column*: reconstruction using the estimated forward model $\hat{\theta}_1$. *4th column*: true θ_1 (blue) and estimated $\hat{\theta}_1$ parameters (red) of the forward model. We display the average angle error and the average shift error. *2nd row*: trained on θ_0 . *3rd row*: trained on a family of operators. *4th row*: using a plug&play prior. The PSNR of the reconstructed image are indicated below each image. The noise level given to the P&P denoiser has been tuned as to yield the best PSNR on the non-blind problem. Other models do not require tuning at evaluation.

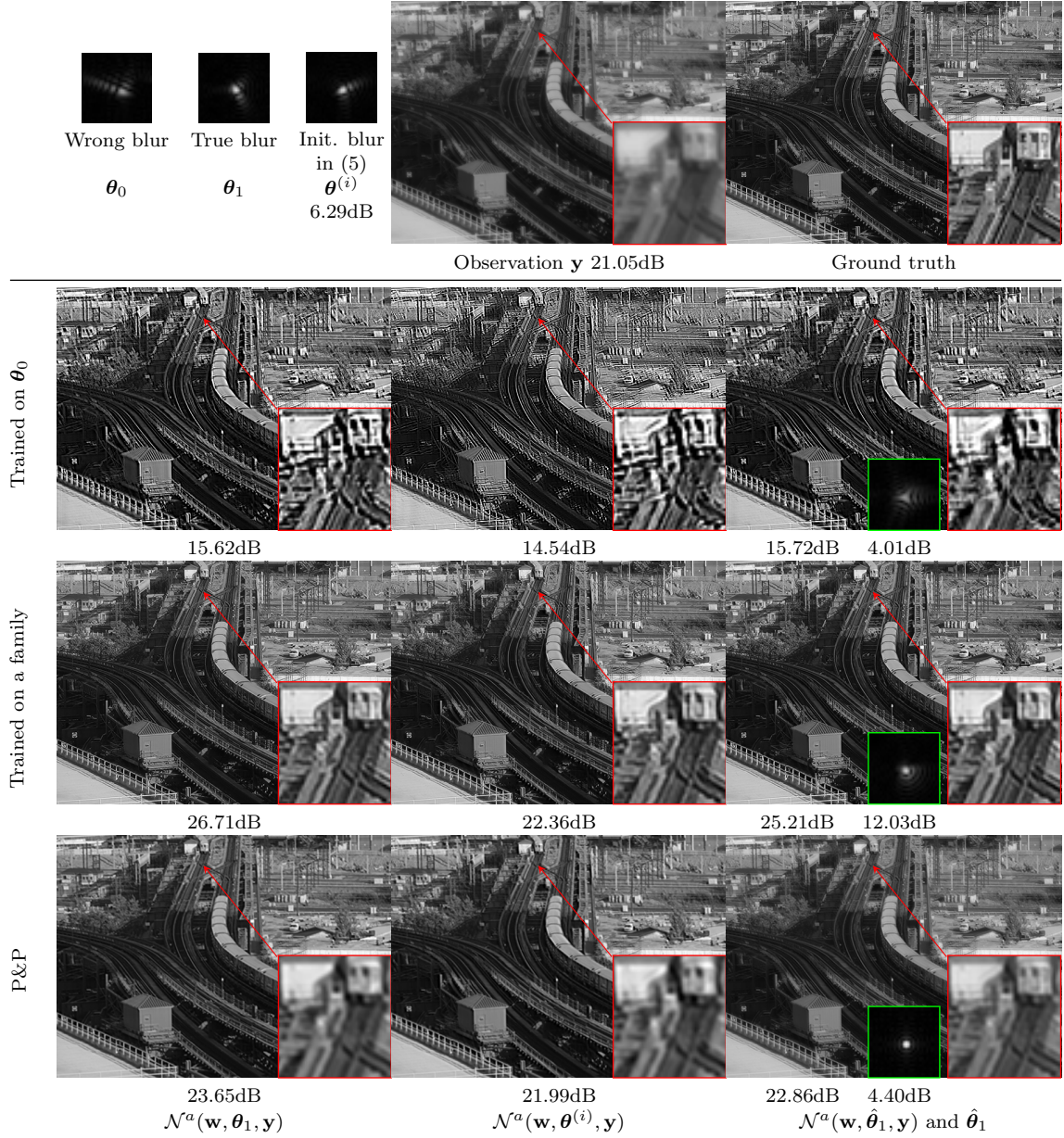


Figure 6: Blind deblurring. *1st column*: reconstruction with a perfect knowledge of the forward model θ_1 . *2nd column*: reconstruction assuming a wrong forward model $\theta^{(i)}$. *3rd column*: reconstruction using the estimated forward model $\hat{\theta}_1$. From top to bottom: different training strategies are compared. *2nd row*: trained on θ_0 . *3rd row*: trained on a family of operators. *4th row*: using a plug&play prior. The PSNR of the reconstructed image and the SNR of the reconstructed blur kernel are indicated below each image. The noise level given to the P&P denoiser has been tuned as to yield the best PSNR on the non-blind problem. Other models do not require tuning at evaluation. The measured signal is $\mathbf{y} = \mathbf{A}(\theta_1)\mathbf{x} + \mathbf{b}$ and the blur of the experiments in the 3rd column is initialized with the blur given in the first row $\theta^{(i)}$.

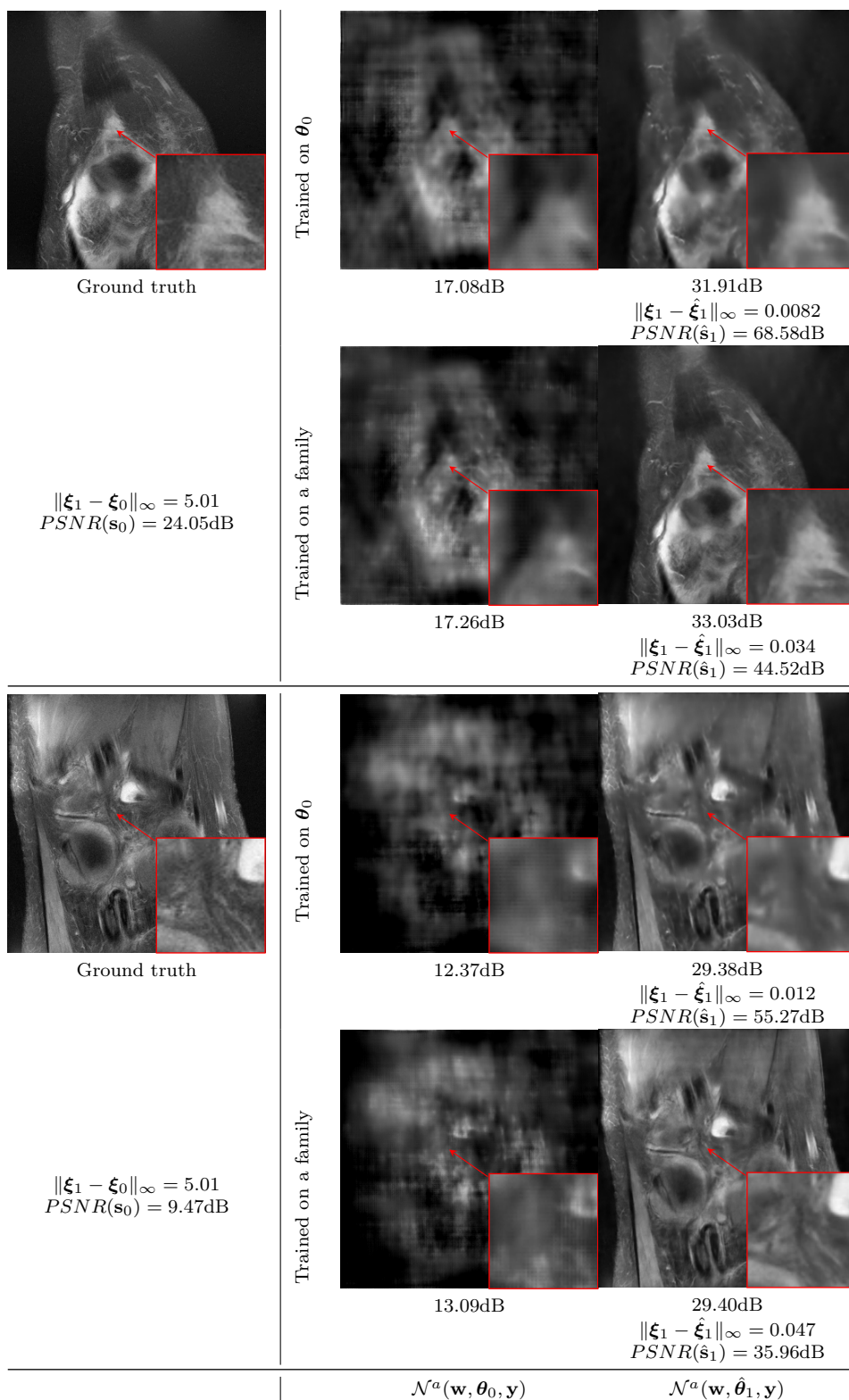


Figure 7: Additional experiments for self-calibrated MRI with different images. *1st column*: ground truth. *2nd column*: reconstruction assuming the wrong forward model θ_0 . *3rd column*: reconstruction using the estimated forward model $\hat{\theta}_1$. From top to bottom for each image: different training strategies are compared. *1st row*: trained on θ_0 . *2nd row*: trained on a family of operators. The PSNR is indicated below each image.

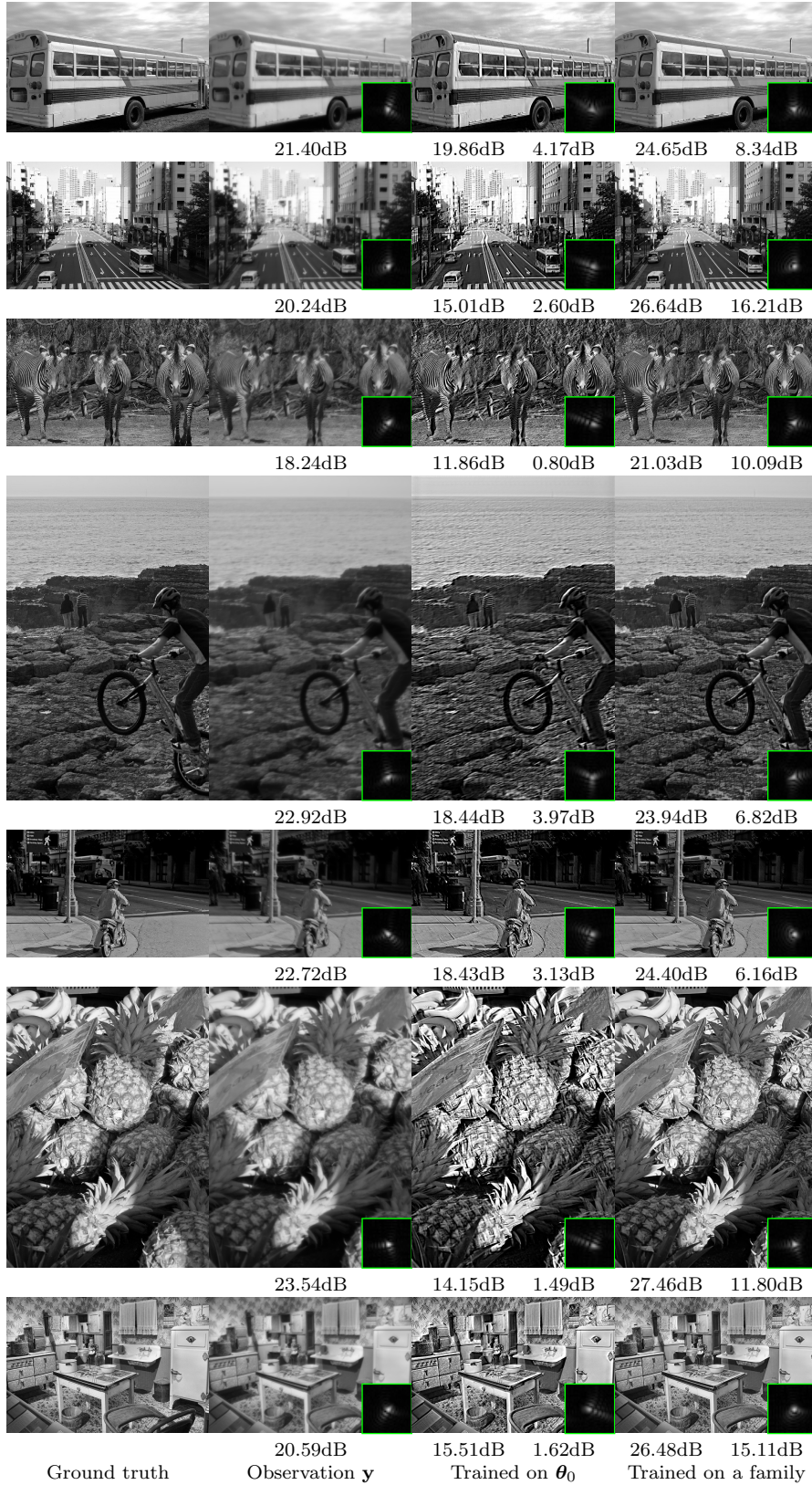


Figure 8: Additional experiments for blind deblurring with different images. *1st column*: ground truth. *2nd column*: observation y and the true blur θ_1 . *3rd column*: reconstruction using the estimated forward model $\hat{\theta}_1$ with the network trained on θ_0 . *4th column*: reconstruction using the estimated forward model $\hat{\theta}_1$ with the network trained on a family of operators. The PSNR of the reconstructed image and the SNR of the reconstructed blur kernel are indicated below each image.

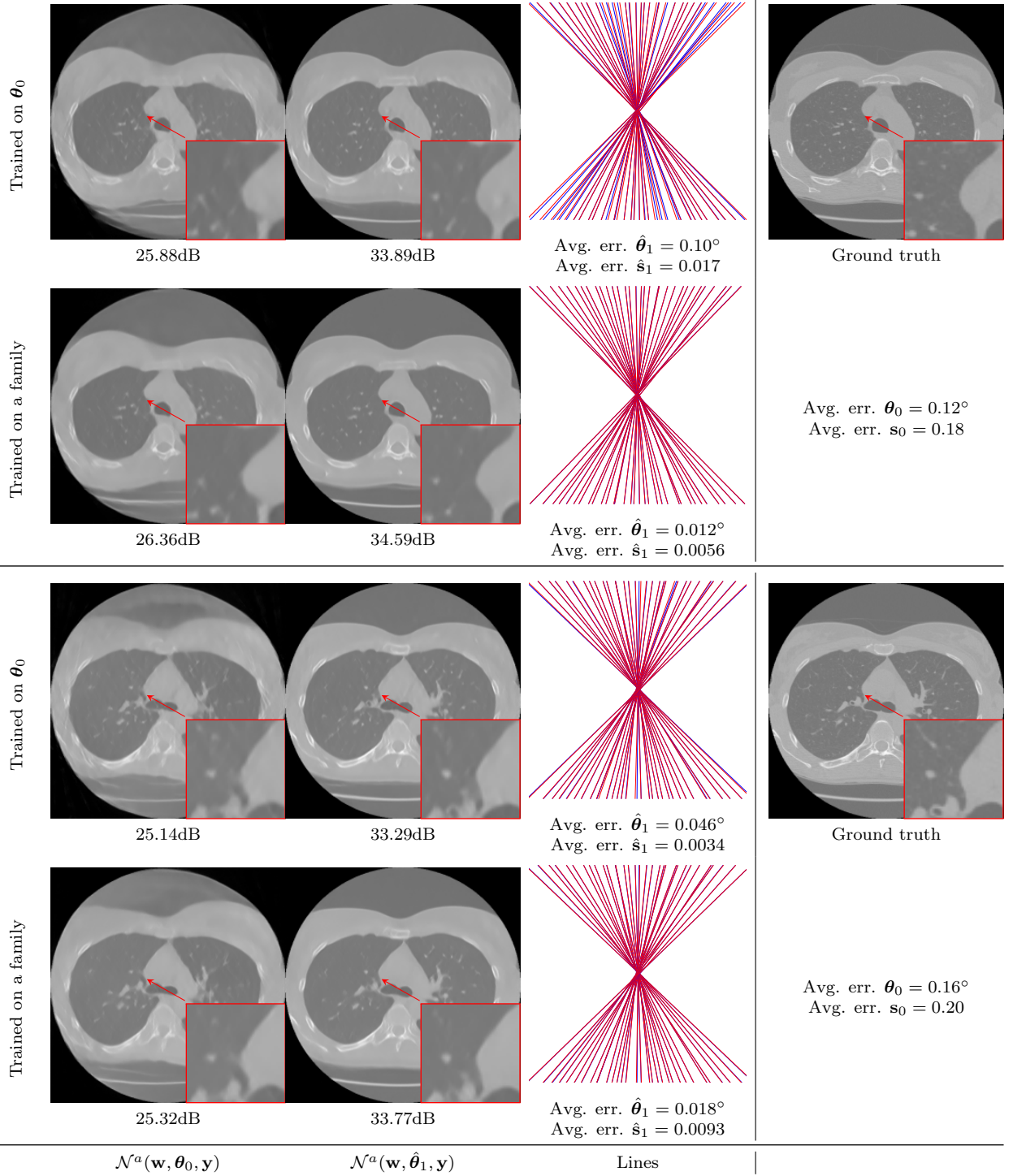


Figure 9: Additional experiments for self-calibrated CT with different images. *1st column*: reconstruction assuming the wrong forward model θ_0 . *2nd column*: reconstruction using the estimated forward model $\hat{\theta}_1$. *3rd column*: true θ_1 (blue) and estimated $\hat{\theta}_1$ parameters (red) of the forward model. We display the average angle error and the average shift error. *4th column*: ground truth. From top to bottom for each image: different training strategies are compared. *1st row*: trained on θ_0 . *2nd row*: trained on a family of operators. The PSNR of the reconstructed image are indicated below each image.

References

- [1] Jonas Adler and Ozan Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12):124007, 2017.
- [2] Jonas Adler and Ozan Öktem. Learned primal-dual reconstruction. *IEEE transactions on medical imaging*, 37(6):1322–1332, 2018.
- [3] Hemant K Aggarwal, Merry P Mani, and Mathews Jacob. Modl: Model-based deep learning architecture for inverse problems. *IEEE transactions on medical imaging*, 38(2):394–405, 2018.
- [4] Ali Ahmed, Benjamin Recht, and Justin Romberg. Blind deconvolution using convex programming. *IEEE Transactions on Information Theory*, 60(3):1711–1732, 2013.
- [5] Vegard Antun, Francesco Renna, Clarice Poon, Ben Adcock, and Anders C Hansen. On instabilities of deep learning in image reconstruction and the potential costs of ai. *Proceedings of the National Academy of Sciences*, 117(48):30088–30095, 2020.
- [6] Samuel G Armato III, Geoffrey McLennan, Luc Bidaut, Michael F McNitt-Gray, Charles R Meyer, Anthony P Reeves, Binsheng Zhao, Denise R Aberle, Claudia I Henschke, Eric A Hoffman, et al. The lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans. *Medical physics*, 38(2):915–931, 2011.
- [7] Simon Arridge, Peter Maass, Ozan Öktem, and Carola-Bibiane Schönlieb. Solving inverse problems using data-driven models. *Acta Numerica*, 28:1–174, 2019.
- [8] Muhammad Asim, Fahad Shamshad, and Ali Ahmed. Blind image deconvolution using deep generative priors. *IEEE Transactions on Computational Imaging*, 6:1493–1506, 2020.
- [9] Yuanchao Bai, Gene Cheung, Xianming Liu, and Wen Gao. Graph-based blind image deblurring from a single photograph. *IEEE transactions on image processing*, 28(3):1404–1418, 2018.
- [10] Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G Dimakis. Compressed sensing using generative models. In *International Conference on Machine Learning*, pages 537–546. PMLR, 2017.
- [11] Emrah Bostan, Reinhard Heckel, Michael Chen, Michael Kellman, and Laura Waller. Deep phase decoder: self-calibrating phase microscopy with an untrained deep neural network. *Optica*, 7(6):559–562, Jun 2020.
- [12] Claire Boyer, Nicolas Chauffert, Philippe Ciuciu, Jonas Kahn, and Pierre Weiss. On the generation of sampling schemes for magnetic resonance imaging. *SIAM Journal on Imaging Sciences*, 9(4):2039–2072, 2016.
- [13] Tatiana A Bubba, Gitta Kutyniok, Matti Lassas, Maximilian März, Wojciech Samek, Samuli Siltanen, and Vignesh Srinivasan. Learning the invisible: a hybrid deep learning-shearlet framework for limited angle computed tomography. *Inverse Problems*, 35(6):064002, 2019.
- [14] Patrizio Campisi and Karen Egiazarian. *Blind image deconvolution: theory and applications*. CRC press, 2017.
- [15] Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509, 2006.
- [16] Ayan Chakrabarti. A neural approach to blind motion deblurring. In *European conference on computer vision*, pages 221–235. Springer, 2016.
- [17] Tony F Chan and Chiu-Kwong Wong. Total variation blind deconvolution. *IEEE transactions on Image Processing*, 7(3):370–375, 1998.
- [18] Nicolas Chauffert, Philippe Ciuciu, Jonas Kahn, and Pierre Weiss. A projection method on measures sets. *Constructive Approximation*, 45(1):83–111, 2017.
- [19] Liang Chen, Faming Fang, Tingting Wang, and Guixu Zhang. Blind image deblurring with local maximum gradient prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1742–1750, 2019.
- [20] David L Colton, Rainer Kress, and Rainer Kress. *Inverse acoustic and electromagnetic scattering theory*, volume 93. Springer, 1998.
- [21] Patrick L Combettes and Jean-Christophe Pesquet. Proximal splitting methods in signal processing. In *Fixed-point algorithms for inverse problems in science and engineering*, pages 185–212. Springer, 2011.
- [22] Valentin Debarnot and Pierre Weiss. Deep-blur: Blind identification and deblurring with convolutional neural networks. 2022.
- [23] Steven Diamond, Vincent Sitzmann, Felix Heide, and Gordon Wetzstein. Unrolled optimization with deep priors. *arXiv preprint arXiv:1705.08041*, 2017.

- [24] Benjamin E Dietrich, David O Brunner, Bertram J Wilm, Christoph Barmet, Simon Gross, Lars Kasper, Maximilian Haeberlin, Thomas Schmid, S Johanna Vannesjo, and Klaas P Pruessmann. A field camera for mr sequence monitoring and system analysis. *Magnetic resonance in medicine*, 75(4):1831–1840, 2016.
- [25] Weisheng Dong, Peiyao Wang, Wotao Yin, Guangming Shi, Fangfang Wu, and Xiaotong Lu. Denoising prior driven deep neural network for image restoration. *IEEE transactions on pattern analysis and machine intelligence*, 41(10):2305–2318, 2018.
- [26] Jean Duchon. Splines minimizing rotation-invariant semi-norms in sobolev spaces. In *Constructive theory of functions of several variables*, pages 85–100. Springer, 1977.
- [27] Rob Fergus, Barun Singh, Aaron Hertzmann, Sam T Roweis, and William T Freeman. Removing camera shake from a single photograph. In *Acm Siggraph 2006 Papers*, pages 787–794. 2006.
- [28] Peter I Frazier. Bayesian optimization. In *Recent advances in optimization and modeling of contemporary problems*, pages 255–278. Informa, 2018.
- [29] Martin Genzel, Jan Macdonald, and Maximilian Marz. Solving inverse problems with deep neural networks—robustness included. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.
- [30] Davis Gilton, Gregory Ongie, and Rebecca Willett. Model adaptation for inverse problems in imaging. *IEEE Transactions on Computational Imaging*, 7:661–674, 2021.
- [31] Amit Goldstein and Raanan Fattal. Blur-kernel estimation from spectral irregularities. In *European Conference on Computer Vision*, pages 622–635. Springer, 2012.
- [32] J.W. Goodman. *Introduction to Fourier Optics*. Electrical Engineering Series. McGraw-Hill, 1996.
- [33] Alban Gossard, Frédéric de Gournay, and Pierre Weiss. Bayesian optimization of sampling densities in mri. *arXiv preprint arXiv:2209.07170*, 2022.
- [34] Alban Gossard, Frédéric de Gournay, and Pierre Weiss. Spurious minimizers in non uniform fourier sampling optimization. *Inverse Problems*, 2022.
- [35] Mark A Griswold, Peter M Jakob, Robin M Heidemann, Mathias Nittka, Vladimir Jellus, Jianmin Wang, Berthold Kiefer, and Axel Haase. Generalized autocalibrating partially parallel acquisitions (grappa). *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 47(6):1202–1210, 2002.
- [36] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng. Weighted nuclear norm minimization with application to image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2862–2869, 2014.
- [37] Kerstin Hammernik, Jo Schlemper, Chen Qin, Jinming Duan, Ronald M Summers, and Daniel Rueckert. σ -net: Systematic evaluation of iterative deep neural networks for fast parallel mr image reconstruction. *arXiv preprint arXiv:1912.09278*, 2019.
- [38] Kyong Hwan Jin, Michael T McCann, Emmanuel Froustey, and Michael Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing*, 26(9):4509–4522, 2017.
- [39] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.
- [40] Dilip Krishnan and Rob Fergus. Fast image deconvolution using hyper-laplacian priors. *Advances in neural information processing systems*, 22, 2009.
- [41] Dilip Krishnan, Terence Tay, and Rob Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR 2011*, pages 233–240. IEEE, 2011.
- [42] Deepa Kundur and Dimitrios Hatzinakos. Blind image deconvolution. *IEEE signal processing magazine*, 13(3):43–64, 1996.
- [43] Carole Lazarus, Pierre Weiss, Nicolas Chauffert, Franck Mauconduit, Loubna El Gueddari, Christophe Destrieux, Ilyess Zemmoura, Alexandre Vignaud, and Philippe Ciuciu. Sparkling: variable-density k-space filling curves for accelerated t2*-weighted mri. *Magnetic resonance in medicine*, 81(6):3643–3661, 2019.
- [44] Bruno Lecouat, Thomas Eboli, Jean Ponce, and Julien Mairal. High dynamic range and super-resolution from raw image bursts. *ACM Trans. Graph.*, 41(4), jul 2022.
- [45] Bruno Lecouat, Jean Ponce, and Julien Mairal. Lucas-kanade reloaded: End-to-end super-resolution from raw image bursts. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 2350–2359. IEEE, 2021.
- [46] Yuelong Li, Mohammad Tofighi, Vishal Monga, and Yonina C Eldar. An algorithm unrolling approach to deep image deblurring. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7675–7679. IEEE, 2019.
- [47] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

- [48] Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1):503–528, 1989.
- [49] Marina Ljubenović and Mário A. T. Figueiredo. Plug-and-play approach to class-adapted blind image deblurring. *International Journal on Document Analysis and Recognition (IJDAR)*, 22(2):79–97, March 2019.
- [50] Michael Lustig, Jin Hyung Lee, David L Donoho, and John M Pauly. Faster imaging with randomly perturbed, under-sampled spirals and ℓ_1 reconstruction. In *Proceedings of the 13th annual meeting of ISMRM*, page 685, Miami Beach, FL, USA, 2005.
- [51] Tomer Michaeli and Michal Irani. Blind deblurring using internal patch recurrence. In *European conference on computer vision*, pages 783–798. Springer, 2014.
- [52] Matthew J Muckley, Bruno Riemenschneider, Alireza Radmanesh, Sunwoo Kim, Geunu Jeong, Jingyu Ko, Yohan Jun, Hyungseob Shin, Dosik Hwang, Mahmoud Mostapha, et al. Results of the 2020 fastmri challenge for machine learning mr image reconstruction. *IEEE transactions on medical imaging*, 40(9):2306–2317, 2021.
- [53] Jinshan Pan, Zhe Hu, Zhixun Su, and Ming-Hsuan Yang. Deblurring text images via l0-regularized intensity and gradient prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2908, 2014.
- [54] Jinshan Pan, Deqing Sun, Hanspeter Pfister, and Ming-Hsuan Yang. Blind image deblurring using dark channel prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1628–1636, 2016.
- [55] Daniel Potts, Gabriele Steidl, and Manfred Tasche. Fast fourier transforms for nonequispaced data: A tutorial. *Modern sampling theory*, pages 247–270, 2001.
- [56] Luc Pronzato. Minimax and maximin space-filling designs: some properties and methods for construction. *Journal de la Société Française de Statistique*, 158(1):7–36, 2017.
- [57] Klaas P Pruessmann, Markus Weiger, Markus B Scheidegger, and Peter Boesiger. Sense: sensitivity encoding for fast mri. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 42(5):952–962, 1999.
- [58] Wenqi Ren, Xiaochun Cao, Jinshan Pan, Xiaojie Guo, Wangmeng Zuo, and Ming-Hsuan Yang. Image deblurring via enhanced low-rank prior. *IEEE Transactions on Image Processing*, 25(7):3426–3437, 2016.
- [59] Nicolai André Brogaard Riis, Yiqiu Dong, and Per Christian Hansen. Computed tomography reconstruction with uncertain view angles by iteratively updated model discrepancy. *Journal of Mathematical Imaging and Vision*, 63(2):133–143, 2021.
- [60] Matteo Ronchetti. Torchradon: Fast differentiable routines for computed tomography. *arXiv preprint arXiv:2009.14788*, 2020.
- [61] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992.
- [62] Ernest Ryu, Jialin Liu, Sicheng Wang, Xiaohan Chen, Zhangyang Wang, and Wotao Yin. Plug-and-play methods provably converge with properly trained denoisers. In *International Conference on Machine Learning*, pages 5546–5557. PMLR, 2019.
- [63] Christian J Schuler, Michael Hirsch, Stefan Harmeling, and Bernhard Schölkopf. Learning to deblur. *IEEE transactions on pattern analysis and machine intelligence*, 38(7):1439–1451, 2015.
- [64] Yu-hsuan Shih, Garrett Wright, Joakim Andén, Johannes Blaschke, and Alex H Barnett. cufinufft: a load-balanced gpu library for general-purpose nonuniform ffts. In *2021 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, pages 688–697. IEEE, 2021.
- [65] Efrat Shimron, Jonathan I Tamir, Ke Wang, and Michael Lustig. Subtle inverse crimes: Naïvely training machine learning algorithms could lead to overly-optimistic results. *arXiv preprint arXiv:2109.08237*, 2021.
- [66] Daniel K Sodickson and Warren J Manning. Simultaneous acquisition of spatial harmonics (smash): fast imaging with radiofrequency coil arrays. *Magnetic resonance in medicine*, 38(4):591–603, 1997.
- [67] Jian Sun, Wenfei Cao, Zongben Xu, and Jean Ponce. Learning a convolutional neural network for non-uniform motion blur removal. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 769–777, 2015.
- [68] Jian Sun, Huibin Li, Zongben Xu, et al. Deep admn-net for compressive sensing mri. *Advances in neural information processing systems*, 29, 2016.
- [69] Tijmen Tieleman and G Hinton. Divide the gradient by a running average of its recent magnitude. coursera neural networks for machine learning. *Mach. Learn.*, 6:26–31, 2012.
- [70] S Johanna Vannesjo, Nadine N Graedel, Lars Kasper, Simon Gross, Julia Busch, Maximilian Haeberlin, Christoph Barmet, and Klaas P Pruessmann. Image reconstruction using a gradient impulse response model for trajectory prediction. *Magnetic resonance in medicine*, 76(1):45–58, 2016.

- [71] Singanallur V Venkatakrisnan, Charles A Bouman, and Brendt Wohlberg. Plug-and-play priors for model based reconstruction. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 945–948. IEEE, 2013.
- [72] Ge Wang, Jong Chu Ye, Klaus Mueller, and Jeffrey A Fessler. Image reconstruction is a new frontier of machine learning. *IEEE transactions on medical imaging*, 37(6):1289–1296, 2018.
- [73] Guanhua Wang, Tianrui Luo, Jon-Fredrik Nielsen, Douglas C Noll, and Jeffrey A Fessler. B-spline parameterized joint optimization of reconstruction and k-space trajectories (bjork) for accelerated 2d mri. *IEEE Transactions on Medical Imaging*, 2022.
- [74] Renke Wang, Roxana Alexandru, and Pier Luigi Dragotti. Perfect reconstruction of classes of non-bandlimited signals from projections with unknown angles. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5877–5881. IEEE, 2022.
- [75] Tomer Weiss, Ortal Senouf, Sanketh Vedula, Oleg Michailovich, Michael Zibulevsky, and Alex Bronstein. Pilot: Physics-informed learned optimal trajectories for accelerated mri. *Journal of Machine Learning for Biomedical Imaging*, 2021.
- [76] Li Xu, Shicheng Zheng, and Jiaya Jia. Unnatural l0 sparse representation for natural image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1107–1114, 2013.
- [77] Ruomei Yan and Ling Shao. Blind image blur estimation via deep learning. *IEEE Transactions on Image Processing*, 25(4):1910–1921, 2016.
- [78] Jure Zbontar, Florian Knoll, Anuroop Sriram, Tullie Murrell, Zhengnan Huang, Matthew J Muckley, Aaron Defazio, Ruben Stern, Patricia Johnson, Mary Bruno, et al. fastMRI: An open dataset and benchmarks for accelerated MRI. *arXiv preprint arXiv:1811.08839*, 2018.
- [79] Jian Zhang and Bernard Ghanem. Ista-net: Interpretable optimization-inspired deep network for image compressive sensing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1828–1837, 2018.
- [80] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte. Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.
- [81] Meina Zhang, Yingying Fang, Guoxi Ni, and Tiejiong Zeng. Pixel screening based intermediate correction for blind deblurring. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 01–09, 2022.
- [82] Bo Zhu, Jeremiah Z Liu, Stephen F Cauley, Bruce R Rosen, and Matthew S Rosen. Image reconstruction by domain-transform manifold learning. *Nature*, 555(7697):487–492, 2018.